



UNIVERSIDADE DO VALE DO TAQUARI - UNIVATES
PROGRAMA DE PÓS-GRADUAÇÃO *STRICTO SENSU*
DOUTORADO EM ENSINO

**O DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL
E AS EVIDÊNCIAS DA ALFABETIZAÇÃO EM CÓDIGO EM ADULTOS**

Walkiria Helena Cordenonzi

Lajeado/RS, agosto de 2020

Walkiria Helena Cordenonzi

**O DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL
E AS EVIDÊNCIAS DA ALFABETIZAÇÃO EM CÓDIGO
EM ADULTOS**

Tese apresentada ao Programa de Pós-Graduação em Ensino da Universidade do Vale do Taquari - Univates, na linha de pesquisa Recursos, Tecnologias e Ferramentas no Ensino, como exigência para a obtenção do título de Doutora em Ensino.

Orientador: Prof. Dr. José Claudio Del Pino

Lajeado/RS, agosto de 2020.

Walkiria Helena Cordenonzi

**O DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL
E AS EVIDÊNCIAS DA ALFABETIZAÇÃO EM CÓDIGO
EM ADULTOS**

A Banca examinadora abaixo aprova a Tese apresentada ao Programa de Pós-Graduação em Ensino da Universidade do Vale do Taquari - Univates, na linha de pesquisa Recursos, Tecnologias e Ferramentas no Ensino, como exigência para a obtenção do título de Doutora em Ensino:

Prof. Dr. José Claudio Del Pino - orientador
Universidade do Vale do Taquari - Univates

Prof. Dr. Edson Prestes e Silva Júnior
Universidade Federal do Rio Grande do Sul

Prof. Dra. Eniz Conceição Oliveira
Universidade do Vale do Taquari - Univates

Prof. Dr. Érico Marcelo Hoff do Amaral
Universidade Federal do Pampa

Prof. Dra. Márcia Jussara Hepp Rehfeldt
Universidade do Vale do Taquari - Univates

Lajeado/RS, 28 de agosto de 2020.

DEDICATÓRIA

Com amor, aos meus filhos

Patricia e Bruno

AGRADECIMENTOS

A minha família, que soube me compreender nos momentos que estava ausente.

Ao meu orientador, que com muita sabedoria, me auxiliou e me amparou nos momentos difíceis e também por sua amizade e paciência.

Aos professores/colegas/amigos que aceitaram ser banca deste trabalho, trazendo grandes contribuições.

A minha amiga e colega Vanessa Mattoso Cardoso, por sempre estar e seu apoio incondicional.

Aos meus amigos, que tiveram a paciência e o carinho de me apoiar e entender as minhas ausências.

Ao colega Robert Gomez, por confiar no meu trabalho e ceder o espaço para a realização do primeiro estudo de caso.

Ao Instituto Federal de Ciência e Tecnologia – IFSUL - por todo o seu apoio, inclusive financeiro.

Enfim, agradeço a todos os seres que de alguma forma e de algum outro plano me auxiliaram e acompanharam nesta caminhada. Minha gratidão!

RESUMO

As tecnologias digitais de informação e comunicação foram surgindo e, à medida que seu uso foi sendo incorporado pelas pessoas, a sociedade foi se tornando diferente, alterando seu modo de funcionamento, hábitos e costumes. Atualmente, muitos cidadãos têm acesso à internet e aos dispositivos móveis, como celulares, e os utilizam para executar suas tarefas tanto particulares, quanto laborais. Essas mudanças compeliram os sujeitos a serem alfabetizados digitalmente. Porém, esses conhecimentos já não estão sendo suficientes, uma vez que as pessoas são encorajadas a produzir tecnologia – e não apenas serem consumidoras –, a fim de satisfazer o mercado de trabalho como profissionais, independente da sua área de atuação. Para um sujeito se tornar agente produtor de tecnologia, preconiza-se que desenvolva habilidades, pelo menos de resolução de problemas, mais precisamente o Pensamento Computacional (PC), em designação cunhada por Wing em 2006. Assim, esta tese tem como objetivo compreender e analisar o desenvolvimento do PC nos sujeitos adultos, os quais já completaram a educação básica, tanto em escolas brasileiras como uruguaias. O escopo deste trabalho se estende para os alunos matriculados em cursos binacionais, no nível pós-médio. Para tanto, foi desenvolvida pesquisa que segue o método indutivo, com abordagem do problema de forma qualitativa, utilizando pesquisa bibliográfica e estudo de caso como procedimentos técnicos. Para desenvolver este tema, foi proposto um Modelo de Referência de Pensamento Computacional (MRPC) e um método de avaliação, além de instanciar um constructo andragógico para validar ambos. O MRPC foi alicerçado na teoria de aprendizagem significativa, proposta por Ausubel (1968), na Andragogia definida por Knowles (1981), além de fazer uso da programação para aplicativos móveis. No MRPC são definidas cinco habilidades básicas, para que o sujeito se aproprie dos conhecimentos, permitindo a sua classificação quanto à alfabetização ou letramento em código, pensador computacional desplugado ou se não há evidências de desenvolvimento do PC, sendo uma das principais contribuições desta tese. Como resultado deste processo, em dois estudos de caso realizados, foi possível determinar que os adultos podem desenvolver as habilidades e competências e que não existem diferenças significativas entre brasileiros e uruguaios quanto à alfabetização em código e o desenvolvimento do pensamento computacional.

Palavras-chave: Pensamento Computacional. Alfabetização em Código. Aprendizagem Significativa. Andragogia.

ABSTRACT

Digital information and communication technologies began to emerge and, as their use was being incorporated by people, society became different, changing its way of functioning, habits and customs. Nowadays, many citizens have access to the internet and to mobile devices, such as cell phones, and use them to perform their particular and work tasks. These changes compelled the subjects to be digitally literate. However, this knowledge is no longer sufficient, since people are encouraged to produce technology - instead of being just consumers -, in order to satisfy the market as professionals, regardless of their area of expertise. For a subject to become a technology-producing agent, it is recommended that they develop skills, at least in problem solving, more precisely Computational Thinking (CT), in designation coined by Wing in 2006. Thus, this thesis aims to understand and analyze the development of CT in adult subjects, who have already completed basic education, both in Brazilian and Uruguayan schools. The scope of this work extends to students enrolled in binational courses, at the post-secondary level. In order to achieve this, a research following the inductive method was developed, with a qualitative and quantitative approach to the problem, using bibliographic research and case studies as technical procedures. To develop this theme, a Reference Model for Computational Thinking (RMCT) and an evaluation method was proposed, in addition to instantiating a andragogic construct to validate both. The RMCT was based on the theory of meaningful learning, proposed by Ausubel (1968), in the Andragogy defined by Knowles (1981), in addition to making use of programming for mobile applications. In the RMCT, five basic skills are defined, so that the subject can appropriate the knowledge, allowing their classification regarding coding literacy, unplugged computational thinker or if there is no evidence of CT development, being one of the main contributions of this thesis. As a result of this process, in two case studies carried out, it was possible to determine that adults can develop skills and habilities and that there are no significant differences between Brazilians and Uruguayans regarding coding literacy.

Keywords: Computational Thinking. Code Literacy. Meaningful Learning. Andragogy.

RESUMEN

Las tecnologías digitales de la información y la comunicación surgen en la medida que su uso fue incorporado por los sujetos. La sociedad se volvió diferente, alterando su modo de funcionamiento, hábitos y costumbres. Actualmente muchos ciudadanos tienen acceso a internet y a dispositivos móviles, como celulares, y los utilizan para ejecutar sus tareas tanto privadas como laborales. Estos cambios forjaron la necesidad de que el sujeto se alfabetizara digitalmente. Pues, esos conocimientos ya no son suficientes, una vez que las personas son requeridas a producir tecnologías y no solo a consumirlas, con la finalidad de cumplir con el mercado laboral como profesionales, independiente del área de actuación de cada uno. Para que un sujeto se convierta en agente productor de tecnología, se requiere que desarrolle habilidades, por lo menos respecto a la resolución de problemas, más precisamente en el Pensamiento Computacional (PC), término acuñado por Wing en 2006. Esta tesis tiene como objetivo comprender y analizar el desarrollo de PC en los sujetos adultos, aquellos que ya completaron educación secundaria tanto en escuelas brasileñas como uruguayas. Este trabajo es extensivo a los alumnos matriculados en cursos binacionales, en el nivel intermedio. Por lo tanto, fue desarrollada una investigación que siguieron el método inductivo, abordando el problema de forma cuanti-cualitativa, utilizando para ello revisión bibliográfica y estudios de caso como procedimientos técnicos. Para desarrollar este tema se propuso un Modelo de Referencia del Pensamiento Computacional (MRPC), y un método de evaluación, además de un constructo andragógico para validar ambos. El MRPC se basa en la teoría de aprendizaje significativo de Ausubel (1968), la Andragogía definida por Knowles (1981), además de hacer uso de la programación para aplicativos móviles. En el MRPC son definidas cinco habilidades básicas para que el sujeto se apropie de los conocimientos, permitiendo su clasificación en cuanto a la alfabetización en código, pensador computacional desenchufado o si no hay evidencias de desarrollo PC, siendo una de las contribuciones principales de esta tesis. Como resultado de este proceso, en dos de los estudios de caso realizados, se pudo comprobar que los adultos pueden desarrollar habilidades y competencias que no existen diferencias significativas entre brasileños y uruguayos en cuanto a la alfabetización en código.

Palabras claves: Pensamiento Computacional. Alfabetización en Código. Aprendizaje significativo. Andragogía.

LISTA DE FIGURAS

Figura 1 - Teoria de Ausubel.....	41
Figura 2 - Aprendizagem significativa e mecânica	42
Figura 3- Formas de Aprendizagem Significativa.....	44
Figura 4 - Representação esquemática do Modelo Ausubeliano.....	48
Figura 5 - Aprendizagem por recepção e descoberta	49
Figura 6 - Maturação natural x Grau de dependência	54
Figura 7 - Modelo da andragogia de Knowles	55
Figura 8 - Contrato de Aprendizagem	59
Figura 9 - Equivalência da Organização dos Sistemas Educacionais Brasil e Uruguai.....	64
Figura 10 - Processo que transforma código-fonte em executável.....	88
Figura 11- Mapa Digital	90
Figura 12 - Níveis de Alfabetização e suas dependências.....	92
Figura 13 - Ferramentas e Alfabetizações	94
Figura 14 - Metodologias utilizadas	98
Figura 15 - Método de estudo de caso.....	101
Figura 16 - Etapas da pesquisa	103
Figura 17 - Modelo MRPC.....	108
Figura 18 – Interface da App Cambio	115
Figura 19 - Tipos de instrumentos avaliativos	117
Figura 20 - Processo de avaliação para o MRPC	118
Figura 21 - Exemplo da questão 2 do pré-teste	119
Figura 22 - Questão 01 – Teste 2.....	120
Figura 23 - Início do pós-teste.....	122
Figura 24 - Questões do pós-teste	123

Figura 25 - Parte da ISO/IEC 25010	124
Figura 26 - Atividades de avaliação de produto de <i>Software</i>	126
Figura 27 - Avaliação do Projeto Final	127
Figura 28 - Logotipo do Curso	131
Figura 29 – Uma aula do EC1	132
Figura 30 - <i>Timeline</i> das atividades do curso	134
Figura 31 – Um encontro no EC1	135
Figura 32 - Código-fonte do App Gato do A02	143
Figura 33 - Código-fonte do App Tradutor do A02	143
Figura 34 - Código-fonte do App Combustível do A02	144
Figura 35- Código-fonte do Projeto Final de A04	145
Figura 36 - Código-fonte do Projeto Final do A06	146
Figura 37 - Código-fonte do Projeto Final do A08	146
Figura 38 - Código-fonte do Projeto Final do A09	147
Figura 39 - Código-fonte do A10	148
Figura 40 - Questão 4 do Pré-teste	152
Figura 41 - Alunos fazendo suas atividades	156
Figura 42 - Intervenção da professora	156
Figura 43 - <i>Timeline</i> das atividades do EC2	157
Figura 44 - Porção de código-fonte App Gato	167
Figura 45 – Parte do código-fonte correto do App Gato	167
Figura 46 - Porção código-fonte do App Combustível	168
Figura 47 - Código-fonte sugerido para o App Combustível	169
Figura 48 - Parte do código-fonte do Projeto Final do A06	170
Figura 49 - App Passos do A17	172
Figura 50 - Parte do código-fonte do Projeto Final do A17	173
Figura 51 - Código-fonte do App Fatorial de A17	173
Figura 52 - Interface do App Fatorial de A17	174
Figura 53 - Código-fonte do App Tradutor do A18	176
Figura 54 - Interface do Projeto Final do A18	177
Figura 55 - Porções do código-fonte do A02	180
Figura 56 - Fragmentos de código-fonte do A20	181
Figura 57 - <i>Feedback</i> do Projeto Final do A20	182
Figura 58 - Fluxograma dos testes	191

Figura 59 - Resumo da avaliação entre BR e UY.....	199
Figura 60 - Fluxograma das correlações.....	202
Figura 61 - Sentido e força da correlação em função do valor de r.....	204

LISTA DE GRÁFICOS

Gráfico 1 - Definições mais utilizadas na definição de PC	34
Gráfico 2 - Teses por áreas	70
Gráfico 3 - Número de Dissertações X Níveis de Ensino	74
Gráfico 4 - Dissertações por área	75
Gráfico 5 - Resumo da avaliação objetiva.....	138
Gráfico 6 - Quantidade de alunos por Parecer.....	149
Gráfico 7 - Resumo dos acertos nos testes (EC2)	161
Gráfico 8 - Acertos x Habilidade (EC2).....	161
Gráfico 9 - Acertos do aluno A18	175
Gráfico 10 - Quantidade de alunos por Parecer do EC2.....	178
Gráfico 11 - Diferença das Médias – Parte 1.....	183
Gráfico 12 - Diferença das Médias – Parte 2.....	184
Gráfico 13 - Sujeitos X Nacionalidade.....	187
Gráfico 14 - Alfabetização x Nacionalidade	187
Gráfico 15 - Histograma da Média de todos sujeitos	190
Gráfico 16 - Gráfico da Distribuição Normal.....	192
Gráfico 17 - Distribuição dos Dados de sujeitos BR e UY	195
Gráfico 18- Histograma dos Dados de BR e UY.....	195
Gráfico 19- <i>Box Plot</i> dos sujeitos BR e UY	197
Gráfico 20 - Assimetrias.....	198
Gráfico 21 - Diagrama de Dispersão entre H1 e H2.....	206
Gráfico 22 - Diagrama de Dispersão entre H1 e H5.....	206
Gráfico 23 - Diagrama de Dispersão entre H3 e H4.....	207

Gráfico 24 - Diagrama de Dispersão entre H3 e H5.....	207
---	-----

LISTA DE QUADROS

Quadro 1- Resumo das Definições sobre PC	33
Quadro 2 - Equivalências das Teorias Ausubeliana e de Adultos.....	60
Quadro 3- Relação dos trabalhos correlatos	69
Quadro 4 - Resumo das Teses X Área.....	73
Quadro 5 - Dados das Dissertações em Ensino	77
Quadro 6 - Resumo Conceitual sobre AD e LD.....	84
Quadro 7- Resumo dos Conceitos e Práticas dos App	113
Quadro 8 - Resumo da avaliação dos App do EC1	142
Quadro 9 - Resumo da avaliação dos App do sujeito A17.....	170
Quadro 10 - Resumo da avaliação dos Apps do A18.....	175
Quadro 11 - Alunos com status LCod	178
Quadro 12- Dados da Amostra	190
Quadro 13 - Dados das habilidades dos sujeitos	200
Quadro 14 – Resultado do Teste de Normalidade.....	202
Quadro 15 - Teste t das Habilidades.....	203
Quadro 16 - Coeficiente de Pearson.....	204
Quadro 17 - Coeficiente de Determinação	204

LISTA DE TABELAS

Tabela 1 - Dados de Teses defendidas a partir de 2009	70
Tabela 2 - Resultado dos testes da avaliação objetiva por habilidades – EC1	137
Tabela 3 - Resultado da ACod (EC1).....	139
Tabela 4 - Avaliação subjetiva	140
Tabela 5 - Resultado dos testes da avaliação objetiva por habilidade – EC2.....	158
Tabela 6 - Resultado da ACod (EC2).....	162
Tabela 7 - Resultado da avaliação subjetiva do EC2.....	164
Tabela 8 - Resumo da avaliação dos App do sujeito A06	166
Tabela 9 - Variação das Médias dos Testes no EC2.....	184
Tabela 10 - Resumo dos dados descritivos de EC1 e EC2.....	188
Tabela 11 - Teste t para duas amostras.....	193
Tabela 12 - Dados Descritivos de BR e UY	194
Tabela 13 - Teste t entre BR e UY	195

LISTA DE ABREVIATURAS

AC	Alfabetização Científica
ACod	Alfabetizado em Código
ACT	Alfabetização Científica e Tecnológica
AD	Alfabetização Digital
APC	Avaliação de Pensamento Computacional
App	Aplicativo
CC	Ciência da Computação
CoP	Comunidade de Prática
CRTC	Currículo de Referência em Tecnologia e Computação
CSTA	<i>Computer Science Teachers Association</i>
CV	Coefficiente de Variação
EB	Ensino Básico
EC1	Estudo de Caso 1
EC2	Estudo de Caso 2
EF	Ensino Fundamental
EM	Ensino Médio
H1	Habilidade de Compreensão
H2	Habilidade de Abstração
H3	Habilidade de Resolução de Problemas
H4	Habilidade de Resolução Algorítmica
H5	Habilidade de Avaliação
ISTE	<i>International Society for Technology in Education</i>
LCod	Letrado em Código

MRPC	Modelo de Referência de Pensamento Computacional
PC	Pensamento Computacional
PCD	Pensador Computacional Desplugado
TDIC	Tecnologias Digitais de Informação e Comunicação
AVA	Ambiente Virtual de Aprendizagem

SUMÁRIO

1 INTRODUÇÃO	19
1.1 Objetivo	21
1.2 Problema e hipóteses	21
1.3 Justificativa	22
1.4 Trajetória acadêmica	24
1.5 Organização da Tese	25
2 REFERENCIAL TEÓRICO	28
2.1 Pensamento Computacional (PC)	28
2.2 Avaliação do PC (APC).....	35
2.3 Teorias da Aprendizagem	39
2.3.1 Aprendizagem significativa	40
2.3.1.1 Conceitos	47
2.3.2 Andragogia	51
2.3.2.1 Contratos de aprendizagem – CAPre	58
2.4 Cursos binacionais.....	62
2.5 Trabalhos correlatos	66
2.5.1 Teses.....	70
2.5.2 Dissertações	73
3 ALFABETIZAÇÃO: CIENTÍFICA, DIGITAL, EM CÓDIGO	79
3.1 Alfabetização Científica	79
3.2 Alfabetização Digital (AD).....	82
3.3 Alfabetização em Código (ACod).....	85
3.4 Considerações e contribuição	95
4 PROCEDIMENTOS METODOLÓGICOS	96
4.1 Caracterização da pesquisa	96
4.2 Desenvolvimento da pesquisa	102
4.2.1 Proposição das habilidades	105
4.2.2 Modelo de referência para o desenvolvimento do PC – MRPC	108
4.2.3 Constructo das aulas	110

4.2.4 Instrumentos de avaliação	116
4.2.4.1 Avaliação objetiva	119
4.2.4.2 Avaliação do produto	124
4.2.4.3 Avaliação Subjetiva	127
5 ESTUDO DE CASO 1 (EC1).....	130
5.1 Resultados e discussão.....	136
6 ESTUDO DE CASO 2 (EC2).....	151
6.1 Resultados e discussão.....	157
6.2 Considerações sobre a quantidade de testes	183
7 RESULTADOS FINAIS E DISCUSSÕES	186
7.1 Coeficiente de Correlação e Determinação	200
8 CONCLUSÃO.....	208
REFERÊNCIAS	213

1 INTRODUÇÃO

O ensino e a aprendizagem, desde muito tempo, vêm sendo foco de pesquisas e experimentações. A educação superior, pelo menos no Brasil, iniciou no século XVII e também vem sendo tema de investigações (CUNHA, 2011). Surgiu a Ciência da Computação por volta da década de 60 do século passado (CERF, 2012), mas ainda hoje não está totalmente assimilada nos currículos escolares. As escolas precisam mudar e realizar a intersecção do conhecimento já instituído com o uso das tecnologias digitais. Esta convergência de temas também não é atual.

A sociedade muda ao longo do tempo. Sua dinâmica é remodelada devido às Tecnologias Digitais de Informação e Comunicação (TDIC), que exercem constantes e permanentes adaptações na vida cotidiana e, com respeito às mudanças, não há mais retorno. É a sociedade permeada pelo mundo digital. A partir dessas transformações, gerou-se uma onda de ajustes e acomodações na comunidade. Isso significa que já se estão percebendo os reflexos nas tarefas diárias, de tal sorte que uma atividade simples não se faz da mesma maneira como se fazia há um tempo atrás.

Sendo assim, aos cidadãos cabe a busca do conhecimento e, juntamente, a alfabetização digital (AD), pois a exploração e a descoberta dessas informações são atividades chave do dia a dia (NAP, 2010). Segundo a definição que consta no livro “Sociedade da Informação no Brasil - Livro Verde”, editado pelo Ministério da Ciência e Tecnologia em 2000, a AD é “a aquisição de habilidades básicas para o uso de computadores e da Internet” (TAKAHASHI, 2000, p. 31). Ademais, é fazer com que os sujeitos sejam agentes capazes de usar as tecnologias podendo tornar sua capacidade laboral mais eficiente e eficaz, além de desenvolver sua capacidade de aprender a aprender.

Wing, em 2006, apresenta uma nova designação: pensamento computacional (PC). Esta expressão pode ser entendida como o processo de resolver e identificar problemas, utilizando

métodos e técnicas provenientes da ciência da computação. Ainda, afirma essa autora, que ao utilizar-se dessas técnicas facilitará a obtenção dos resultados, principalmente para encontrar soluções de problemas complexos. Não somente esta interpretação sobre PC, como também sugere que seja compreendido como um conjunto de habilidades e apropriação de determinados processos cognitivos. Na literatura, encontram-se diferentes definições sobre esta expressão, e, em virtude disso, ainda hoje não há um consenso sobre o assunto. Posto isso, este trabalho seguirá a conceituação de Wing (2017).

De tal forma, o PC e suas habilidades correspondentes são indispensáveis para os profissionais da área de Ciência da Computação (CC). No entanto, Wing (WING, 2008) afirma que estas habilidades são proveitosas para todas as pessoas, independente da profissão ou domínio de conhecimento. Sobretudo, para facilitar o empoderamento dos profissionais quando submetidos a resolver problemas em qualquer área, mesmo que aparentemente estas não estejam relacionadas.

Portanto, incorporar o ensino de PC na educação formal, no início da educação, se torna muito importante, para que os alunos já comecem a dominar o PC e conseqüentemente a resolução de problemas. Em alguns países, já existem movimentos nesse sentido, principalmente na educação primária, como, por exemplo, no Reino Unido, em que foi criada uma disciplina “Computing” obrigatória a partir de 5 até 16 anos de idade (THE ROYAL SOCIETY, 2012).

Enquanto não se adequam os currículos para auxiliar o desenvolvimento do PC, muitos investigadores propuseram cursos (oficinas), nos quais desenvolvem atividades nesse sentido, alguns focando na programação, outros em tarefas desplugadas (BRACKMANN, 2017). É certo que para programar é necessário o desenvolvimento de certas habilidades que agregam a aprendizagem de algoritmos e, ainda, uma linguagem de programação. Em outras palavras, desenvolver o PC na programação não significa apenas programar, mas sim desenvolver todo o processo para encontrar a solução de um problema.

Desse modo, une-se a preocupação dos processos de ensino e aprendizagem para analisar como os sujeitos adultos desenvolvem seu pensamento computacional, tanto para atender as demandas da sociedade, quanto para lograr resolver problemas do cotidiano. Assim sendo, associa-se a preocupação das mudanças causadas pelo advento das tecnologias, por parte de sujeito adulto que irá estar ou já está no mercado de trabalho (RUIPÉREZ, 2017), e concomitantemente buscando sua qualificação com as demandas de incorporação de novas habilidades, inclusive de pensamento computacional, se traçou o objetivo deste trabalho.

1.1 Objetivo

Esta pesquisa tem por objetivo compreender e analisar o desenvolvimento do PC nos sujeitos adultos (que já completaram a educação básica), com formação em escolas brasileiras e uruguaias.

Os sujeitos desta pesquisa são os alunos que atualmente estão matriculados em um curso binacional e pós-médio, com a finalidade de englobar os cursos técnicos no nível subsequente e os cursos superiores. Contextualizando dessa maneira, qualquer curso a partir do Ensino Médio poderá ser considerado. Já a questão da formação em escolas brasileiras ou uruguaias se dá pela nacionalidade do sujeito. Esta diferenciação está calcada nos cursos binacionais, para os quais metade das vagas são ofertadas para brasileiros e a outra metade para uruguaios. Estes cursos nasceram de um acordo binacional entre os governos brasileiro e uruguaio, representado pelo Instituto Federal de Ciência e Tecnologia Sul-Rio-Grandense (IFSUL) e pelo *Consejo de Educación Técnico Profesional – Universidad del Trabajo del Uruguay* (CETP-UTU), respectivamente. O que também caracteriza esses cursos é a dupla diplomação. Inegavelmente, este contexto de trabalho é singular, a começar pelo uso de dois idiomas em sala de aula.

Os objetivos específicos são:

- a) verificar o processo de alfabetização em código (*code literacy*) dos sujeitos. A partir do levantamento dos conhecimentos prévios do aluno, averiguar o nível de alfabetização digital (AD) e em código, pois apoiado nesses dados o professor começará a traçar o conhecimento do aluno sobre quais habilidades já possui e quais logrará desenvolver;
- b) propor um modelo de referência e um método de avaliação de PC para adultos;
- c) a partir das interações de ensino e aprendizagem sobre PC, analisar e classificar a alfabetização em código dos sujeitos;
- d) analisar as habilidades e competências dos sujeitos com formação no Brasil e no Uruguai, em uma situação de ensino e aprendizagem de PC.

Os dois primeiros objetivos serão respondidos no Capítulo 4 e os dois últimos, respectivamente nos Capítulos 5, 6 e 7, por meio da proposição de uma metodologia de ensino inovadora baseada na teoria de aprendizagem significativa de Ausubel (AUSUBEL; NOVAK; HANESIAN, 1983), na andragogia (KNOWLES, 1981) e na programação de aplicativos móveis.

1.2 Problema e hipóteses

O problema da pesquisa é: como se dá o desenvolvimento do pensamento computacional (PC) de alunos adultos com educação pós-médio binacional (brasileira e uruguaia)? As hipóteses apresentadas a seguir surgem de observações da autora, aliadas aos resultados de outras pesquisas (GIL, 2002, p. 35), a fim de analisar o processo de aprendizagem de PC de um indivíduo adulto:

- a) O sujeito pode desenvolver as habilidades (ACod) de PC.
- b) O sujeito pode desenvolver competências¹ (LCod) de PC.
- c) O sujeito pode desenvolver um projeto de *software* utilizando PC.²
- d) Há diferenças no resultado de formação do PC entre os sujeitos com formações diferentes de ensino, ou seja, entre brasileiros e uruguaios.

1.3 Justificativa

Na perspectiva acadêmica, muitos estudos de casos foram realizados, nos quais se trabalham o desenvolvimento cognitivo do PC e de programação nas escolas, a partir da educação básica, em virtude da alta demanda de profissionais na área de tecnologia. De certo, os resultados mostram melhorias em várias disciplinas (MORENO LEÓN; ROMÁN GONZÁLEZ; ROBLES, 2018), mas ainda faltam ferramentas para avaliar esta habilidade.

Muitos pesquisadores (GROVER; PEA, 2013; KOH *et al.*, 2010) concordam que ainda é necessário entender como auxiliar os alunos a desenvolverem o PC, melhorando suas habilidades de resolução de problemas, e como deve ser o processo de aprendizagem, pois essas habilidades são fundamentais para os cidadãos do século XXI.

Neste trabalho, o olhar está voltado para os indivíduos adultos. Escolheu-se a fase adulta para a pesquisa porque, primeiramente, é o público com quem a autora atua e, ao fazer leituras exaustivas sobre o tema, a grande maioria das pesquisas, conforme pode ser corroborado na Seção 2.5 dos trabalhos correlatos, acontecem com os sujeitos que se encontram na educação básica. Além da importância tanto mercadológica quanto pessoal do indivíduo, de este ser produtor de tecnologia e não somente consumidor, isto poderia ser uma das principais motivações para desenvolver o PC.

¹ A competência será observada a partir da análise dos códigos fonte desenvolvidos pelo sujeito. A partir do ponto de vista do sujeito é desenvolver uma solução para um problema de forma diferente do qual aprendeu ou observou – uma inovação.

² O projeto de *software* compreende todo o ciclo de vida, independente da complexidade do produto a ser desenvolvido.

Uma pesquisa colaborativa da Google com a empresa de consultoria norte-americana McKinsey & Company³ com o propósito de mapear as habilidades do brasileiro quanto ao uso de ferramentas e tecnologias digitais, através do *Digital Skills Index* (Índice de Habilidades Digitais), com uma escala que varia de 0 a 5, obteve nota 3. Nesta foi concluída que:

[...] essa nota reflete que o brasileiro domina hoje muito bem as habilidades mais básicas, relacionadas a acesso, mas ainda tem grandes lacunas em competências mais técnicas, como o uso das ferramentas para a produção de conteúdo e programação. A melhora nessas atividades mais sofisticadas poderia trazer efeitos positivos no aumento da produtividade, na maior participação no mercado de trabalho, na geração de renda e na redução do desemprego (GOOGLE, 2019, texto digital, grifo nosso).

Como se pode perceber por esse relato, é evidente para os adultos a necessidade de desenvolver as habilidades de PC, bem como a programação. Portanto, um indivíduo que compreende os princípios computacionais será capaz de transpor um algoritmo para qualquer linguagem de programação, pois este é uma base que independe de linguagens. As ferramentas mudam, mas a essência do PC permanece.

O acesso fácil à internet, a popularização dos dispositivos móveis, principalmente os *smartphones*, e o uso de vários aplicativos disponíveis nas lojas virtuais tornam os usuários consumidores de tecnologia. Porém, estes mesmos atores podem ser capazes de ser produtores de tecnologia; portanto, deverão ser, pelo menos, alfabetizados em código. Rushkoff (2012, tradução nossa, texto digital), preocupado com o uso dos programas, sem entender o que estava acontecendo no ambiente digital, resolve alfabetizar as crianças, argumentando que a “alfabetização em código é um requisito mínimo para participar de um ambiente digital”. Em 2009, Resnick *et al.* denominavam esse conhecimento como “fluidez digital” e seu trabalho centrou-se em ensinar crianças (a partir de 8 anos) e jovens a produzir código através de uma ferramenta chamada Scratch⁴. Porém, esses autores não descreveram nenhum método, independente da linguagem de programação, para avaliar e definir a alfabetização em código.

Para complementar, existem poucos trabalhos de PC com adultos, ignorando que uma das profissões do futuro será a de programador. Inquestionavelmente, os *softwares* estão em todo

³ Reconhecida como a líder mundial no mercado de consultoria empresarial global e de gestão que atende empresas líderes, governos, organizações não governamentais e organizações sem fins lucrativos. Disponível em: <https://www.mckinsey.com/br/careers/impact-experience/sobre-a-mckinsey>. Acesso em: 5 set. 2019.

⁴ É um projeto do Lifelong Kindergarten Group, do MIT Media Lab. *Scratch* é uma linguagem gráfica de programação, inspirada no Logo, que possibilita a criação de histórias interativas, animações, simulações, jogos e músicas. Também permite compartilhamento *online* dos projetos desenvolvidos na plataforma. Disponível em: <http://www.scratchbrasil.net.br/index.php/sobre-o-scratch.html>. Acesso em: 12 jun. 2018.

lugar, desde aplicativos de deslocamento (como UBER®⁵, por exemplo), nos eletrodomésticos (Internet das coisas⁶), ou seja, as aplicações estão presentes em diferentes nichos de mercado, sendo que o mais importante para as empresas é investir no desenvolvimento da lógica nos profissionais que apresentam mais alto desempenho; embora esses tenham um maior custo, o investimento é feito sobre a inteligência humana, e não em uma máquina (RAINIE; ANDERSON, 2017).

1.4 Trajetória acadêmica

Aos 22 anos, eu me tornei bacharel em Informática pela PUC-RS. No ano seguinte, 1993, iniciei como professora de Informática na Escola Técnica da Universidade Regional Integrada do Alto Uruguai e das Missões (URI), Câmpus Santo Ângelo. Durante dois anos, atendi os alunos do Ensino Básico – da 1ª à 8ª série, trabalhando com LOGO. Esta é uma ferramenta desenvolvida há mais de 30 anos por Papert (1980a), para ensinar as crianças a programar (e desenvolver o PC). Refletindo na escrita desta tese, percebo que desde aquela época já estava envolvida nos temas em que continuo interessada: ensino e aprendizagem, pensamento lógico, algoritmos, programação e resolução de problemas.

Depois, passei a atuar no Ensino Superior. Na mesma Instituição, no Curso de Administração de Empresas com ênfase em Análise de Sistemas (já extinto), continuo com a disciplina de Algoritmos I (basicamente com os mesmos conteúdos). Paralelamente, fiz uma pós-graduação, e o tema desenvolvido no trabalho final foi avaliar a qualidade de um produto de *software* na área educacional. Neste tempo, despertei para o tema de qualidade. Em 1998, ingressei no Mestrado em Ciências da Computação na UFRGS. O tema da dissertação foi propor um modelo de qualidade de produto de *software* e de processo de desenvolvimento de *software* para um *Workflow*.

Entre a conclusão do Mestrado e a decisão de seguir pesquisando passaram-se 18 anos, nos quais sempre atuei em sala de aula, em diferentes instituições de ensino, entre elas a

⁵ A Uber Technologies Inc. é uma empresa que possui uma plataforma de tecnologia que conecta motoristas parceiros com usuários por meio de um aplicativo para *smartphones*. Disponível em: <https://help.uber.com/pt-BR/riders/article/como-a-uber-funciona?nodeId=738d1ff7-5fe0-4383-b34c-4a2480efd71e>. Acesso em: 05 set. 2019.

⁶ Em 1999, Kevin Ashton, do MIT, propôs a expressão “Internet das Coisas” e pode ser compreendida como um conjunto de objetos conectados à internet e entre si através da rede, munidos de sensores, circuitos eletrônicos e *softwares* e capazes de coletar, processar e trocar dados. Disponível em: <http://finep.gov.br/noticias/todas-noticias/4446-kevin-ashton-entrevista-exclusiva-com-o-criador-do-termo-internet-das-coisas>. Acesso em: 20 ago. 2019.

Universidade Luterana do Brasil, Câmpus de Palmas (Tocantins); Centro Universitário Franciscano (hoje, Universidade Franciscana) e atualmente no Instituto Federal de Educação, Ciência e Tecnologia Sul-Rio-Grandense (IFSUL), Câmpus Santana do Livramento. O meu ingresso neste último aconteceu em 2010, trabalhando desde então com a disciplina de Lógica e Algoritmos, além de outras disciplinas concomitantemente. Esta última fase foi decisiva para a escolha do tema e do problema da pesquisa. Ao longo desses oito anos, sendo responsável por esta disciplina, no Curso Técnico de Informática para Internet, no nível pós-médio, me aportaram alguns conhecimentos e também muitas dúvidas. Muitas experiências e pesquisas foram realizadas, principalmente com a docente que atua na área de Matemática, mais precisamente na disciplina de Fundamentos Matemáticos da Computação. Juntas, muitos artigos foram apresentados em eventos e também em parcerias com colegas de outras Instituições, sempre buscando soluções para melhorar os processos de ensino e aprendizagem relacionados ao tema de algoritmos e resolução de problemas.

Ao fazer os cálculos do tempo, foram-se 25 anos atuando nesta área: pensando em como melhorar a prática docente, como melhorar ou alterar o processo avaliativo, refletindo sobre acertos e erros, por que os alunos não aprendem, o que fazer para conseguir sua atenção, entre muitas outras preocupações. Conforme sugere Minayo (2002), as inquietações de pesquisa nascem do cotidiano, mas o que a impulsiona é o novo e o confronto com o desconhecido. Espero que pelo menos uma destas inquietações possa ser respondida no final deste trabalho, o que, para mim, já terá sido de grande importância.

1.5 Organização da Tese

Este trabalho está organizado em capítulos, em que já no primeiro, de caráter introdutório, apresenta uma visão global do tema, dos objetivos, do problema de pesquisa e hipóteses, da justificativa e organização das demais divisões.

No Capítulo 2, apresenta-se o referencial teórico sobre os temas julgados importantes e afins com esta pesquisa: diferentes entendimentos e conceituação sobre PC e como avaliar. Também os pressupostos andragógicos que embasam este trabalho: aprendizagem significativa e andragogia são apresentados, além de um breve histórico e entendimento dos cursos binacionais. Na continuidade, os trabalhos correlatos são discutidos, daqueles que interseccionam alguns dos temas envolvidos nesta tese.

No Capítulo 3, está delineado um estudo que inicia na compreensão sobre Alfabetização Científica e Tecnológica, perpassando pelas várias formas de alfabetizações e, por fim, define-se a diferença entre alfabetização em código (ACod) e letramento em código (LCod), e junto ao modelo apresentado no capítulo seguinte os requisitos mínimos para inferir ao indivíduo o seu conhecimento para ser considerado ACod.

No Capítulo 4, a pesquisa está detalhadamente retratada. A partir do problema observado, ou seja, o ponto de partida, foi utilizado o método indutivo, abordando-se o problema de forma quali-quantitativa, e utilizando-se de procedimentos técnicos como pesquisa bibliográfica e estudo de caso. Apresenta-se o levantamento das habilidades básicas necessárias para desenvolver o PC, com foco no adulto, se descreve o Modelo de Referência de Pensamento Computacional (MRPC), no qual as habilidades são listadas e definidas, sendo elas: compreensão, abstração, resolução de problema, resolução algorítmica e avaliação. A resolução algorítmica envolve conhecimentos de programação. Seguindo o modelo proposto por Brennan e Resnick (2012), que definem conceitos e práticas, neste trabalho definem-se os conceitos julgados pertinentes. Já para a prática, propôs-se um curso denominado de “Eu Programo 1.0!”, no qual são desenvolvidos aplicativos (App) que incorporam os conceitos de programação e sua prática para dispositivos móveis. Neste ponto, poderá o aluno avançar em suas habilidades de programação e tornar-se um indivíduo alfabetizado em código. A motivação está em conseguir que o sujeito alcance o *status* de ACod.

Esse curso para alunos está calcado nas teorias de Aprendizagem Significativa, de Ausubel (1968), e na Andragogia (aprendizagem de adultos), proposta por Knowles na década de 1970 (KNOWLES; HOLTON; SWANSON, 2005). O primeiro autor resume sua teoria deste modo: conheça e valorize o que o aluno já sabe, e sempre busque o significado de qualquer conceito novo para a realidade do sujeito. Já o segundo autor enfatiza como os adultos aprendem, o que lhes interessa. A união de ambas as teorias foram importantes pilares norteadores para a definição de todos os constructos metodológicos desta tese. Ademais, ao ser realizado e findado o curso, os dois últimos objetivos específicos serão respondidos. Em suma, os procedimentos metodológicos são descritos, bem como algumas contribuições inéditas da tese.

O caso piloto (ou estudo de caso 1) está narrado no Capítulo 5, de forma detalhada e com as discussões sobre os resultados qualitativos encontrados. E no capítulo consecutivo, o segundo estudo de caso é igualmente dissertado.

No Capítulo 7, são apresentados os dados quantitativos de ambos os estudos e as conclusões pertinentes.

A seguir, as considerações finais estão registradas, acompanhadas das referências e dos apêndices citados no texto. Todo o constructo andragógico definido no Capítulo 4 está disponibilizado no Ambiente Virtual de Aprendizagem (AVA), neste endereço: <http://ecompartindo.santana.ifsul.edu.br/course/index.php?categoryid=22>.

2 REFERENCIAL TEÓRICO

O objetivo deste capítulo é explanar e analisar as definições, os conceitos, as práticas (entre outros) dos diversos materiais e/ou recursos que estão disponíveis na literatura sobre os temas nos quais esta pesquisa foi ancorada. Cada tema julgado importante se tornou um subcapítulo. Ao discorrer sobre esses assuntos e os autores que os discutem, permitiu-se, também, inferir o entendimento na esfera desta tese. Tais subcapítulos são denominados como Pensamento Computacional (PC), Avaliação do PC, Teorias da Aprendizagem, Cursos Binacionais e trabalhos correlatos.

2.1 Pensamento Computacional (PC)

Há diversos entendimentos e definições acerca do conceito de PC (do inglês, *Computational Thinking*), porém ainda não há um consenso sobre o que é PC ou o que ele implica. Nesse sentido, Wing (2006, p. 33, tradução nossa) define essa expressão como sendo uma habilidade que “envolve a resolução de problemas, projeção de sistemas e compreensão do comportamento humano, através da extração de conceitos fundamentais da ciência da computação”. Além disso, a autora acrescenta que é uma forma de pensamento recursivo, usando a abstração e a decomposição, ou seja, é fazer uso do raciocínio heurístico para encontrar a solução. Ainda, defende que é uma habilidade fundamental a qual poderá ser utilizada por todas as pessoas do século XXI, mostrando-se semelhante à de ler, escrever e calcular. Em outras palavras, Wing (2006, p. 33, tradução nossa) expressa que o PC é:

Um método que humanos, não computadores, pensam. Pensamento computacional é uma maneira para seres humanos resolverem problemas; não é os seres humanos pensarem como computadores. Computadores são tediosos e chatos; humanos são inteligentes e imaginativos. Nós humanos tornamos a computação empolgante.

Equipados com dispositivos computacionais, usamos nossas habilidades para resolver problemas que não ousaríamos tentar antes da era da computação e construir sistemas com funcionalidades restritas apenas pela nossa imaginação.

Partindo do exposto por Wing (2006), o PC refere-se às pessoas e como estas desenvolvem a capacidade de resolver problemas, independentemente dos recursos computacionais. Esta autora sugere que todas as universidades deveriam ter no currículo uma disciplina como “Formas de pensar como um cientista da computação”, não somente iniciando com atividades e tarefas de programação, mas para todos, independentemente do curso ou idade do sujeito.

Os conceitos de PC podem ser incorporados e utilizados de forma transversal nas disciplinas, tanto da educação básica, como da educação superior, através de técnicas de resolução de problemas (YADAV *et al.*, 2014). Estes mesmos estudiosos, concordando com as ideias propostas por Wing, enfatizam que é importante expor os estudantes ao desenvolvimento do PC, no início de seus estudos, auxiliando-os a “[...] tornarem-se conscientes sobre quando e como aplicar essas ideias [...]” (YADAV *et al.*, 2014, p. 2, tradução nossa). As reflexões apresentadas nas publicações desses autores estão concentradas no nível K-12⁷.

Uma vez que o PC não está mais restrito somente a profissionais de CC, este trabalho o toma como importante para todas as pessoas (comunidade), a fim de que possam desenvolver também essa habilidade de pensamento.

Retomando Wing (2006, p. 33), a autora definia o PC como “pensar como um cientista da computação”. Já em 2017 redefiniu este conceito como sendo: “O pensamento computacional são os processos de pensamento envolvidos na formulação de um problema e em expressar sua(s) solução(ões) de tal forma que um computador – humano ou máquina – possa efetivamente executá-los” (WING, 2017, p. 8, tradução nossa), sendo que a abstração é a habilidade de mais alto nível dentro do processo de PC, como também salienta que a solução pode ser implementada por humanos ou máquinas.

A *International Society for Technology in Education* (ISTE), associação que representa mais de 100.000 profissionais em todo o mundo, em 2016, lançou o *e-book* “*Standards for Students*”. O objetivo dos padrões ISTE é servir de guia para aprender, ensinar e conduzir, na era digital, pessoas que tenham entre 4 e 18 anos, já que neste documento estão propostas atividades variadas, entre elas, de desenvolvimento de PC (ISTE, 2016). A entidade define que o PC consiste em “[...] desenvolver e empregar estratégias para compreender e resolver problemas de

⁷ Nível pré-universitário nos EUA.

maneira a promover o poder dos métodos tecnológicos para desenvolver e testar soluções” (ISTE, 2016, p. 5, tradução nossa).

A *Computer Science Teachers Association* (CSTA) recomenda algumas habilidades para desenvolver o PC, que são: abstração, pensamento algorítmico, modelagem, escala, e reconhecimento de padrões. Essa Associação complementa que há inúmeros trabalhos que corroboram com as habilidades de abstração e de pensamento e desenvolvimento de algoritmos. Além das habilidades já citadas, adiciona que desenvolver o PC é a capacidade de manipular representações simbólicas (YADAV *et al.*, 2015).

Ambas as associações, ISTE e CSTA, esforçam-se para definir padrões e currículos, a fim de desenvolver e aplicar, através de propostas de cursos, o PC, influenciando muitas mudanças curriculares. Em seus trabalhos, o nível de ensino ao qual se debruçam corresponde, no Brasil, ao Ensino Básico. Como exemplo disso, pode-se citar, dentre muitos, o da Inglaterra que alterou seu currículo para agregar o PC (THE ROYAL SOCIETY, 2012).

Mais recentemente no Brasil, em 2018, com a proposta do Currículo de Referência em Tecnologia e Computação (CRTC) definida pelo Centro de Inovação para a Educação Brasileira (CIEB), percebe-se que já existe um movimento referente à inserção do PC na Educação Básica. Este currículo de referência, destinado à Educação Infantil e ao Ensino Fundamental, aponta que “cada uma das habilidades do Currículo de Referência em Tecnologia e Computação se relaciona com uma ou mais competências gerais e habilidades da BNCC⁸” (CRTC, 2018, p. 22) e define o PC como sendo o processo que compreende “sistematizar, representar, analisar e resolver problemas” (CRTC, 2018, p. 19). Seu principal objetivo é prover um suporte curricular para a comunidade educativa nos temas relacionados à tecnologia e à computação. O currículo está organizado em três eixos: Cultura Digital, Tecnologia Digital e Pensamento Computacional. Recomenda que, para a aplicação do último eixo, o docente deverá ter um nível de conhecimento mais avançado na área de CC.

Em sua tese de doutorado, Román González (2016) enfatiza que a conceituação sobre PC ainda não está estabelecida (BRENNAN; RESNICK, 2012; KORKMAZ; ÇAKIR; ÖZDEN, 2017; KOTSOPOULOS *et al.*, 2017; ROMÁN GONZÁLEZ; PÉREZ GONZÁLEZ; JIMÉNEZ FERNÁNDEZ, 2017), não havendo consenso de como incorporar este tipo de pensamento no currículo escolar, nem como avaliá-lo. Este autor apresenta as definições de PC divididas em quatro grupos: genéricos, operacionais, psicológico-cognitivos e educativos curriculares. No primeiro grupo, enquadra o entendimento de Wing (2006, 2008), já descrito anteriormente, e

⁸ Base Nacional Comum Curricular (BNCC)

também cita a seguinte interpretação: “O pensamento computacional é o processo de reconhecer aspectos da computação no mundo que nos rodeia e aplicar ferramentas e técnicas da Ciência da Computação para entender e pensar sobre sistemas e processos naturais e artificiais” (THE ROYAL SOCIETY, 2012, p. 29, tradução nossa).

Na classificação do grupo denominado como operacionais, cita as definições da Fundação Nacional para a Ciência (*National Science Foundation*)⁹, da ISTE e CSTA, todas localizadas nos EUA. Estas últimas definiram PC (para utilizá-la com os educandos na faixa de K-12), como sendo:

O pensamento computacional é um processo de solução de problemas que inclui (não limitando) as seguintes características:

- a) Elaboração de problemas de uma forma que nos permita usar um computador e outras ferramentas para ajudar a resolvê-los;
- b) Organização e análise lógica de dados;
- c) Representação de dados através de abstrações tais com modelos e simulações;
- d) Automatização de soluções através do pensamento algorítmico (uma série de passos ordenados);
- e) Identificação, análise e implementação de possíveis soluções com o objetivo de alcançar a combinação mais eficiente e eficaz das etapas e recursos e
- f) Generalização e transferência do processo de resolução de problemas para um amplo tipo de problemas (ISTE; CSTA, 2011, texto digital, tradução nossa).

Segundo a classificação de Román González (2016), a definição da Google é do tipo psicológico-cognitivo. Como exemplo disso, este grupo de pesquisadores entende deste modo:

O pensamento computacional é um processo de solução de problemas que contém uma série de características, como ordenar e analisar logicamente dados e criar soluções usando uma série de etapas ordenadas (ou algoritmos) e disposições, como a capacidade de lidar com a complexidade e problemas em aberto. O PC é essencial para o desenvolvimento de aplicativos de computador, mas também pode ser usado para apoiar a resolução de problemas em todas as disciplinas, incluindo matemática, ciências e humanidades (GOOGLE FOR EDUCATION, 2015, texto digital, tradução nossa).

Analogamente às definições sobre PC já apresentadas, os estudiosos da Google concordam que o PC é um procedimento utilizado para solução de problemas. Ademais, salientam que pode ser empregado de forma transversal no currículo, ou seja, em qualquer área de conhecimento.

Para o último grupo – Educativos Curriculares – são citadas duas referências, a partir de Román González (2016): a primeira, o modelo CAS (*Computing at School*) do Reino Unido, e a segunda, o modelo proposto por Brennan e Resnick (2012), no MIT¹⁰ Media Lab, desenvolvido para jovens. A partir de sua experiência com o PC, utilizando o *Scratch* em comunidades *on-line*, observações, entrevistas e cursos, os últimos autores definem o PC através de um *framework* que contém três dimensões: “conceitos computacionais (os conceitos são empregados conforme são

⁹ Disponível em: <https://apstudents.collegeboard.org/courses/ap-computer-science-principles>.

¹⁰ MIT - *Massachusetts Institute of Technology*.

programados), práticas computacionais (desenvolvidas enquanto são programadas) e perspectivas computacionais (perspectivas se formam sobre o mundo ao seu redor e sobre si mesmos)” (BRENNAN; RESNICK, 2012, p. 3, tradução nossa).

Na primeira dimensão, um conjunto de sete conceitos computacionais são apresentados: sequências, laços, condições, operadores, paralelismo, eventos e dados. Na dimensão das práticas de desenvolvimento, foram propostos quatro conjuntos: incremental e iterativo, teste e depuração, reutilização e remixagem, e abstração e modularização. As dimensões de conceitos e práticas foram utilizadas como referência para o desenvolvimento do constructo desta tese.

Com base nesta classificação, Román González (2016, p. 163, tradução nossa) apresenta a sua definição sobre PC, classificando-a como operativa e focada nos conceitos computacionais: “O PC é a capacidade de formular e solucionar problemas apoiando-se nos conceitos fundamentais da computação e usando a lógica inerente as linguagens de programação: sequenciais, laços, condições, funções e variáveis”.

Sendo assim, pode-se perceber que esse autor também fundamenta seu trabalho em Brennan e Resnick (2012), citando na definição não apenas os comandos necessários (e básicos) das linguagens de programação, mas também o conceito e uso de variáveis, ou seja, focando o PC, basicamente, como a solução sendo um programa de computador.

É importante ressaltar que a comunidade educativa (BRENNAN; RESNICK, 2012; ROMÁN GONZÁLEZ, 2016; THE ROYAL SOCIETY, 2012), empresas (GOOGLE FOR EDUCATION, 2015), governos (NCE, 2013) que apresentaram definições e entendimentos sobre PC o fizeram considerando os níveis correspondentes à Educação Básica (EB)¹¹ no Brasil.

Além das definições já apresentadas, Brackmann (2017, p. 31) define PC como sendo:

[...] uma distinta capacidade criativa, crítica e estratégica humana de saber utilizar os fundamentos da Computação, nas mais diversas áreas do conhecimento, com a finalidade de identificar e resolver problemas, de maneira individual ou colaborativa, através de passos claros, de tal forma que uma pessoa ou uma máquina possam executá-los eficazmente.

Em virtude disso, esse autor define mais detalhadamente o conceito proposto por Wing em 2006, quando trabalha com o PC de forma desplugada, isto é, sem o uso do computador.

Posteriormente, expõe-se uma síntese de definições sobre PC, a partir dos diversos autores já mencionados, a qual está elencada no Quadro 1. Sua finalidade foi de buscar palavras correlatas, com a intenção de destacar as que são relevantes no escopo desta tese.

¹¹ A Educação Básica compreende o Ensino Fundamental, o qual é composto de 9 anos, e Ensino Médio, com duração de 3 anos. Os estudantes iniciam com 6 anos e a previsão de conclusão é aos 17 anos de idade.

Quadro 1- Resumo das Definições sobre PC

Conceitos e/ou habilidades	(WING, 2008)	(ISTE; CSTA, 2011)	(BRENNAN; RESNICK, 2012)	(CRTC, 2018)	(RIBEIRO; FOSS; CAVALHEIRO, 2017)	(GOOGLE FOR EDUCATION, 2015)
Resolução de Problema	X	X	X	X		X
Abstração	X	X	X	X	X	
Teste	X	X	X		X	
Algoritmo		X	X	X	X	X
Tratamento de Dados		X	X		X	X
Programação		X			X	X
Padrões		X		X		
Decomposição				X		

Fonte: Da autora (2019).

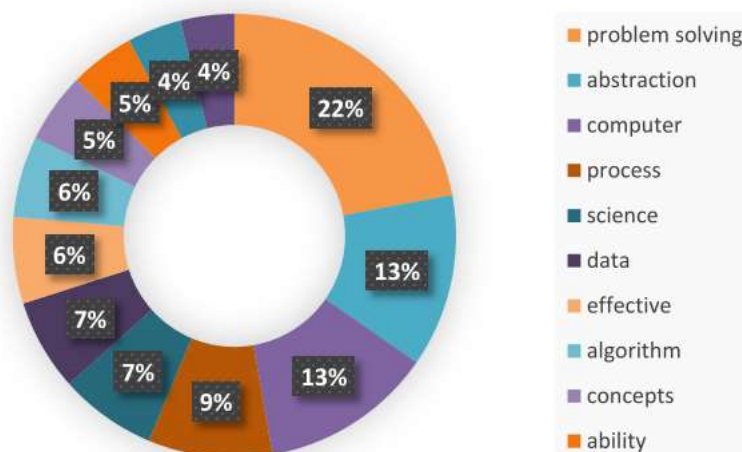
Conforme pode ser observado no Quadro 1, as habilidades de maior concordância são “resolução de problemas”, “abstração” e “algoritmos”. A abstração pode ser considerada como uma das mais destacadas habilidades (RIBEIRO; FOSS; CAVALHEIRO, 2017), sendo indiscutível sua importância para o desenvolvimento do PC (GROVER; PEA, 2013). Não se pode confundir PC com programação, pois esta é a automação das abstrações do indivíduo (WING, 2008).

Neste trabalho, os itens apresentados no Quadro 1 são denominados de habilidades, que Wing denomina de “skill”. Assim sendo, tais itens são tomados como habilidades que o indivíduo deve desenvolver no processo de aprendizagem do PC, apoiado também por Minchillo (2018), sendo este também o entendimento nesta tese.

Reforçando sobre as habilidades mais importantes, Avila *et al.* (2017) analisaram 58 artigos e chegaram à mesma conclusão, ou seja, aos mesmos resultados: as habilidades mais trabalhadas foram pensamento algorítmico, resolução de problemas e abstração.

Kalelioglu, Gülbahar e Kukul (2016) analisaram 125 artigos, recuperados de fontes digitais e bases de dados, examinando a definição, escopo e base teórica sobre pensamento computacional (PC). Sobre o levantamento dos conceitos mais utilizados para descrever o significado de PC, os autores apresentaram o seguinte Gráfico 1.

Gráfico 1 - Definições mais utilizadas na definição de PC



Fonte: Kalelioglu; Gülbahar; Kukul (2016, p. 589).

As informações expressas no Gráfico 1 estão de acordo com os dados mostrados no Quadro 1, no que se refere aos conceitos mais utilizados, que são resolução de problemas (com o maior percentual) e abstração (em segundo lugar). Contudo, neste estudo, algoritmos aparecem em sexto lugar e habilidade, em décimo lugar. A partir desses dados pode-se, novamente, inferir que ainda não há consenso sobre o que é PC.

No tocante a isso, Ribeiro, Foss e Cavalheiro (2017), diferentemente de Wing (2006), denominam os itens (coluna Conceitos e/ou Habilidades do Quadro 1) como pilares do PC. Esse também é o entendimento de Brackmann (2017) e CRTC (2018). Na sua pesquisa, Ribeiro, Foss e Cavalheiro (2017) propõem três pilares: abstração, automação e análise. No relatório do CRTC (2018), foram propostos 4 pilares para o PC: abstração, algoritmos, decomposição e reconhecimento de padrões, que são semelhantes aos citados por Brackmann (2017).

Neste trabalho, conforme já explicitado, o PC é composto por 5 habilidades: compreensão, abstração, resolução de problemas, resolução algorítmica e avaliação. Suas definições estão descritas no Capítulo 4, por serem conceitos importantes nesta tese.

A ideia de resolução de problemas, algoritmos e programação pode ser remetida aos trabalhos de Papert (1980b), nos anos de 1980, cujo autor definiu a linguagem LOGO, na qual as crianças programavam através do movimento de uma tartaruga. Inspirado na teoria construtivista de Piaget, depois de ter trabalhado com ele no Centro de Epistemologia Genética de Genebra, entre os anos de 1958 e 1963, afirmou que “a criança inteligente ensina o computador burro, ao invés de o computador inteligente ensinar a criança burra” (PAPERT, 1980a, p. 9). Sendo considerado o precursor da teoria sobre o construcionismo, Papert afirmava que as crianças são construtoras do seu próprio conhecimento.

Papert (1980a) lançou a proposta de programação e desenvolvimento de código para crianças. Mais tarde, Wing (2006) retoma esta proposta, descrevendo-a como PC, porém de uma forma mais ampla, sem mencionar idade ou faixas etárias.

Em suma, pode-se destacar que esta seção da tese foi elaborada com o intuito de apresentar uma amostra de vários trabalhos sobre o PC e suas aplicações. Como se pode perceber, não é uma ideia nova, mas tomou importância baseado da publicação de Wing em 2006. A partir deste marco, nota-se a presença de movimentos para a incorporação do PC desde os primeiros anos de escola, e modificações nos currículos escolares têm sido propostos e testados, tanto por cientistas e educadores, como empresas de TI (tecnologia da informação) e governos.

Retomando a pesquisa de Kalelioglu, Gülbahar e Kukul (2016), do total de artigos que apresentaram o nível da população estudada, somente 24,8% deles estão no Ensino Superior. Faz-se importante citar este trabalho porque os sujeitos envolvidos são adultos (matriculados em cursos pós-médio) e não há muitos trabalhos que os consideram.

Poucas são as experiências e publicações que apresentam o processo de ensino ou de aprendizagem de PC no nível superior, comparado com a EB (ensino básico). Entende-se que é importante para os adultos desenvolver o PC, a fim de aportar soluções a sua carreira profissional e – por que não – pessoal. É possível desenvolver as habilidades de Pensador Computacional em adultos para torná-los produtores de tecnologia e, por conseguinte, produtores de inovação.

A seguir será exibido o processo de Avaliação do PC.

2.2 Avaliação do PC (APC)

Após a publicação de Wing (2006), muitos trabalhos têm sido desenvolvidos na área de pensamento computacional (PC), com diferentes abordagens, conforme já explicitado. Este tópico tem por objetivo discorrer sobre a preocupação de avaliar o PC. Desde 2011 (BRACKMANN, 2017; VALENTE, 2016), os currículos de alguns países têm sido modificados, a fim de desenvolver nos estudantes, com idades que variam entre 4 e 14 anos, habilidades para que adquiram o *status* de “pensador computacional”¹² (BRENNAN; RESNICK, 2012; ISTE; CSTA, 2011).

Não há um padrão ou modelo genérico para a mensuração do PC. Torna-se emergente que métodos avaliativos sejam aplicados, a fim de que se possa conhecer os resultados do

¹² Do inglês “computational thinker”.

desenvolvimento do PC, mediante as mudanças curriculares que estão sendo propostas e já implementadas.

A seguir, estão descritos alguns trabalhos encontrados na literatura sobre avaliação de PC. Estes foram selecionados a partir da busca no Portal de Periódicos da Capes, respeitando as palavras avaliação (*assessment*), pensamento computacional (*Computational Thinking*) e educação de adultos (*higher education*).

Korkmaz, Çakir e Özden (2017) definiram uma escala que determina os níveis de Habilidade de Pensamento Computacional (*computational thinking skills*) e aplicaram em alunos da Amasya University, na Turquia. Esta escala está dividida em 29 itens, distribuídos em fatores (criatividade, pensamento algorítmico, cooperação, pensamento crítico e resolução de problemas). Aplicaram para 1306 estudantes e concluíram que, estatisticamente, este método é válido e confiável. Este trabalho é uma das poucas referências sobre APC em estudantes que já concluíram a EB. Ainda assim, os autores não descreveram como o método foi aplicado e não apresentam dados de quanto os estudantes desenvolveram as habilidades.

O grupo de Koh *et al.* (2010) desenvolveu uma ferramenta denominada de CTPG (*Computational Thinking Pattern Graph*) para a avaliação semântica de jogos e simulações, desenvolvidos por estudantes, que analisa os conceitos de pensamento computacional implementado pelos alunos. Aplicaram em vários grupos de estudantes universitários, justificando que as linguagens visuais não são adequadas para se obter de forma clara o tipo de conhecimento do qual o aluno se apropriou.

Uma experiência desenvolvida na Universidade de Montana (VAN DYNE; BRAUN, 2014) descreve um curso sobre PC (que possui o mesmo nome), para alunos matriculados no Curso de Ciência da Computação. A finalidade foi de melhorar a resolução de problemas de forma analítica e obtiveram sucesso nos resultados. Concluíram que o curso teve um impacto significativo na melhoria das habilidades de resolução de problemas. Para avaliação, utilizaram o *Whimbey Analytical Skills Inventory* (WASI)¹³ em dois momentos, como pré e pós testes.

Um teste realizado para avaliar o desenvolvimento das habilidades do PC foi desenvolvido e aplicado em estudantes do primeiro semestre de um Curso de Computação na Universidade de Rhodes (África do Sul). O planejamento seguiu a Taxonomia de Bloom e o teste foi aplicado para avaliar os saberes dos estudantes antes de os alunos iniciarem um conhecimento formal dos conteúdos. As questões foram retiradas de *Computer Olympiad Talent Search*¹⁴.

¹³ Desenvolvido por Arthur Whimbey e Jack Lochhead, para avaliar e desenvolver capacidades cognitivas para a resolução de problemas.

¹⁴ Disponível em: <http://www.olympiad.org.za/talent-search>. Acesso em: 3 ago. 2018.

Concluíram que o PC não é uma habilidade que possa ser desenvolvida nos alunos, antes deve ser melhor abordada. Sendo assim, sugerem que mais estudos deveriam ser realizados (GOUWS; BRADSHAW; WENTWORTH, 2013).

No trabalho de Avila *et al.* (2017), os autores apresentam uma revisão sistemática da literatura, entre os anos de 2011 e 2016, sobre avaliação do PC. Dos 58 artigos considerados, 65% foram classificados em intervenção com avaliação própria e intervenção com avaliação própria fundamentada em métodos existentes. Porém, não consideram em sua pesquisa o nível de ensino nos quais os testes foram aplicados, tampouco diferenciaram a avaliação realizada com o uso de ferramentas. Cabe ressaltar que quando os testes são realizados por automação, eles ficam limitados ao *software* utilizado para o desenvolvimento da solução do problema, como, por exemplo, o teste *Dr. Scratch* (descrito a seguir).

Recentemente, Moreno León, Román González e Robles (2018) apontaram que os métodos mais utilizados pela comunidade educativa são *CT-Test*, *Bebras* e *Dr. Scratch*, os quais são empregados para avaliar o PC sob diferentes pontos de vistas e que devem ser utilizados de forma conjunta, sendo compatíveis e complementares.

O primeiro foi proposto na tese de doutorado de Román González (2016), nomeado de *Test de Pensamiento Computacional (CT-Test)*, foi projetado, especificamente, para a educação espanhola que cursa o final do ciclo básico e secundário (idade do aluno entre 10 e 16 anos ou seu correspondente K-12)¹⁵. Este mede o nível de desenvolvimento do PC por meio de um questionário composto por 28 questões objetivas, com opções de 4 respostas (somente uma correta). Concluiu que o PC não só é uma habilidade cognitiva de solução de problemas, como é independente de sexo. Assinalaram, ainda, que o teste é limitado porque não é possível avaliar e medir processos cognitivos como “aplicar” e “assimilar” o PC.

*Dr. Scratch*¹⁶ é uma aplicação *Web* que analisa automaticamente os projetos desenvolvidos no *Scratch*. Fornece *feedback* que permite ao projetista melhorar seu código, aprender com seus erros e desenvolver suas habilidades de PC (MORENO LEÓN; ROBLES; ROMÁN GONZÁLEZ, 2015; MORENO LEÓN; ROMÁN GONZÁLEZ; ROBLES, 2018). No entanto, esta ferramenta limita-se às atividades desenvolvidas exclusivamente no ambiente *Scratch* e não oferece suporte para abstração de funções e procedimentos no código (GROVER; PEA, 2013). O *Scratch* foi desenvolvido para ser utilizado entre as idades de 8 e 16 anos, embora seus criadores afirmam que pode ser empregado em qualquer idade (RESNICK *et al.*, 2009).

¹⁵ No Brasil corresponde ao ensino básico, nos EUA a “middle school”.

¹⁶ Disponível em: <http://www.drscratch.org/>. Acesso em: 16 maio 2018.

Como resultado, esta ferramenta apresenta um retorno que indica a qualidade do programa desenvolvido no *Scratch*, sendo considerada uma ferramenta de avaliação formativa (ROMÁN GONZÁLEZ; PÉREZ GONZÁLEZ; JIMÉNEZ FERNÁNDEZ, 2017).

O *International Challenge on Informatics and Computational Thinking*¹⁷, denominado de *Bebras*¹⁸, é uma iniciativa internacional para promover a computação e pensamento computacional entre os alunos em todas as idades. É um desafio que consiste em um conjunto de questões (chamadas de tarefas) que tem por finalidade testar o nível de desenvolvimento das habilidades do PC (DAGIENÉ; FUTSCHEK, 2008), entre elas: a resolução de problemas, decomposição, projeto de algoritmos, reconhecimento de padrões, generalização e abstração. As tarefas são a parte essencial do desafio e cada uma delas pode abordar conceitos de informática ou de PC (DAGIENÉ; SENTANCE; STUPURIENÉ, 2017). Os desafios são separados por idade: *Pre-Primary* (entre 5 e 8 anos), *Primary* (8 e 10), *Benjamin* (10 e 12), *Cadets* (12 e 14), *Junior* (entre 14 e 16) e *Senior* (idade acima de 16 anos). Os participantes são geralmente supervisionados por professores que podem integrar o desafio Bebras em suas atividades de ensino. Em 2018, o Bebras foi aplicado com 2.614.000 participantes de 45 países. No entendimento de Moreno León, Román González e Robles (2018), este teste pode ser aplicado para qualquer faixa etária.

O *Bebras* está sendo citado aqui, pois muitas questões que o compõem foram utilizadas nos testes avaliativos que constituem o constructo andragógico desta tese. Por outro lado, é importante ressaltar que esta prova surgiu para ser utilizada em competições, e não como um método de avaliação de PC. Portanto, ao ser utilizado, está medindo habilidades diferentes do *CT-Test* (WIEBE *et al.*, 2019), visto que, neste último, suas questões são baseadas, mais especificamente, em programação de blocos.

Muitas outras pesquisas sobre APC foram encontradas (KATAI; TOTH, 2010; KORDAKI, 2010; MORENO, 2012; MORENO LEÓN; ROBLES; ROMÁN GONZÁLEZ, 2015; RODRIGUES *et al.*, 2015; SEITER; FOREMAN, 2013; TOWHIDNEJAD *et al.*, 2015), mas não estão descritas, pois os níveis de ensino que abordaram não é o semelhante ao do escopo desta tese.

Já as pesquisas de Lye e Koh (2014) e de Cavalcante, Costa e Araújo (2016) e Araújo (2019) são mencionadas, visto que ambas utilizaram no seu processo de avaliação o *framework* de Brennan e Resnick (2012) (que também foi fonte para a proposta de avaliação desta tese).

¹⁷ Disponível em: <https://www.bebbras.org/>. Acesso em: 10 jul. 2018.

¹⁸ Bebras em lituano significa Castor.

Ambos os trabalhos abrangeram o *framework* nas 3 dimensões – conceito, prática e perspectiva. O primeiro, de Lye e Koh (2014), realizaram seus olhares no nível K-12. Concluíram que, a partir da resolução de problemas e utilizando-se do construcionismo para desenvolver um artefato concreto, auxiliou no desenvolvimento do PC. Já Cavalcante, Costa e Araújo (2016), utilizaram o *framework* em crianças de EF (Ensino Fundamental) e EM (Ensino Médio), utilizando a plataforma Code.org e programação em blocos.

Em síntese, o campo de APC ainda está carente de pesquisas. Foi possível analisar que boa parte das metodologias propostas são do tipo qualitativas (RAABE *et al.*, 2017), que há poucas ferramentas para avaliação que produzem *feedback* imediato e são restritas ao ambiente de desenvolvimento utilizado. Ainda mais que o desenvolvimento de metodologias de aplicação e avaliação do PC para crianças e jovens (que cursam o EF) contam com um maior referencial teórico e prático se comparado ao nível universitário ou adultos, afirmam Tang *et al.* (2020). Na pesquisa realizada por esses autores, a partir de 77 artigos anteriores a agosto de 2019, apenas 15% estão focadas no Ensino Superior e concluem que ainda há poucas publicações na área de PC, sugerindo a integração de várias ferramentas para melhorar a avaliação da aprendizagem sobre PC.

Cabe esclarecer que o método proposto por Brennan e Resnick (2012), de avaliação do desempenho do PC, não foi utilizado neste trabalho, porque é dependente da ferramenta *Scratch* (ferramenta que não foi considerada no planejamento do curso descrito no Capítulo 4). Enquanto isso, o *Bebras* foi apresentado porque algumas questões de APC utilizadas nesta tese foram retiradas deste teste.

Outro ponto a ser considerado, com relação às teorias de aprendizagem relacionadas na literatura já citada, é que somente Lye e Koh (2014) basearam seu trabalho no construcionismo de Papert, os demais autores não fizeram referências.

A seguir, as teorias de aprendizagem que serviram como sustentação a esta tese.

2.3 Teorias da Aprendizagem

Com o passar do tempo, muitas mudanças aconteceram e novas transformações surgirão, incluindo os aspectos em geral da sociedade, inclusive a educação. Apesar de algumas controvérsias sobre o início da era da informação, não há como negar sua existência, a exemplo disso estão: as tecnologias, a computação móvel e ubíqua, entre outros. É preciso não somente estar atento ao mercado de trabalho, suas necessidades e tendências, mas também aos atuais

estudantes que enfrentarão novas habilidades e competências exigidas aos profissionais. Certamente, o ofício de uma profissão já não é executado da mesma maneira como era há 20 anos, por exemplo.

Conforme já se vem observando, o desenvolvimento do PC, embora remeta a preocupação desde o século anterior, especialmente nos últimos 20 anos está tomando forma, exigindo mudanças de pensamento e atitudes no professor, no aluno, no currículo e nos materiais instrucionais.

Nesta seção, estão descritas as teorias de aprendizagem que ancoram esta tese, levando-se em consideração o desenvolvimento do PC e as habilidades recomendadas nesta pesquisa (e citadas na seção 4.2.2) e está dividida em duas partes. Na primeira, são apresentados os pontos relevantes da teoria de aprendizagem de David Ausubel – chamado de Aprendizagem significativa; e a segunda com a finalidade de comentar os principais aspectos da teoria de aprendizagem de adultos, proposta por Knowles (1973), denominada de Andragogia. O objetivo destas seções não é a apresentação exaustiva de ambas as teorias, mas sim dos aspectos relevantes que são utilizados para embasar esta pesquisa.

2.3.1 Aprendizagem significativa

Ausubel (1968) propõe uma teoria cognitivista que se preocupa com os mecanismos internos da mente, cujo principal conceito é a aprendizagem significativa. Ele entende que a estrutura do conhecimento está organizada de forma hierárquica no sujeito, sendo um conjunto de conceitos, de ideias ou proposições que fazem parte de uma área de conhecimento e que estão interacionadas.

O sujeito (ou aprendiz) aprende quando ancora um novo conhecimento em sua estrutura cognitiva. O fator mais importante na aprendizagem é o conhecimento prévio do aluno. Ausubel resume sua teoria na seguinte frase: “o fator mais importante que influencia a aprendizagem é o que aluno já sabe. Descubra isso e ensine-o de acordo” (AUSUBEL; NOVAK; HANESIAN, 1983, p. 1).

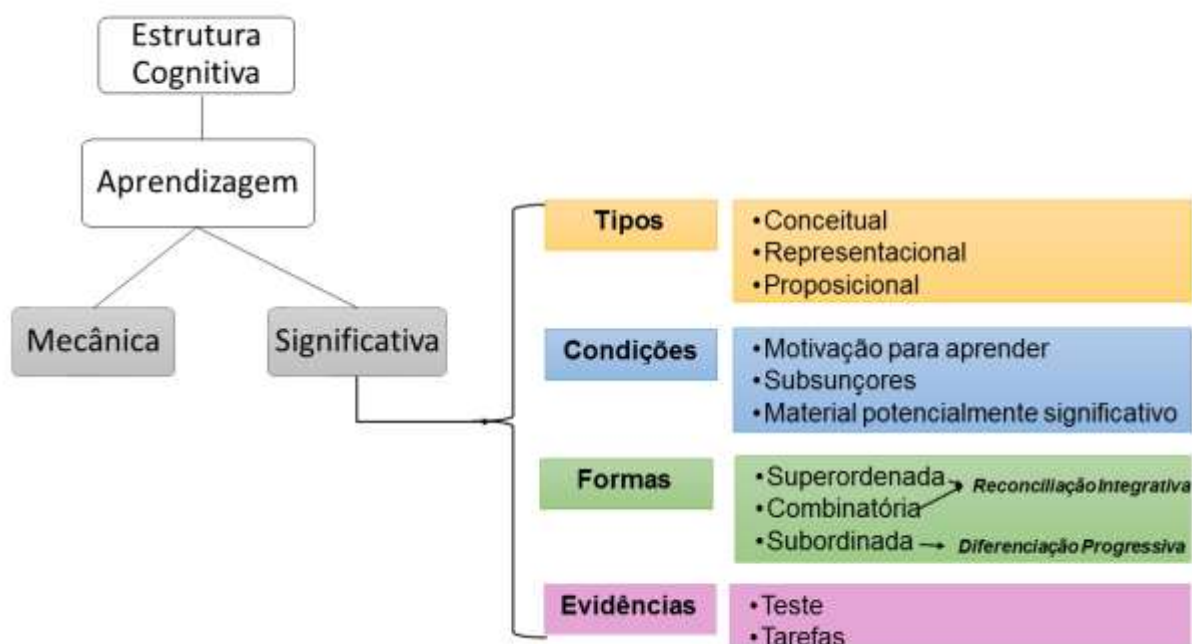
Esse processo ocorre por meio do qual uma nova informação interage de forma não-literal e não arbitrária. Não-literal significa que a relação entre o material a ser aprendido e a estrutura cognitiva não deve ser ao pé da letra (ou seja, a relação não se altera se símbolos diferentes, mas equivalentes, forem usados). Já por não arbitrária entende-se que o significado está na relação do

novo item com a estrutura cognitiva, o qual não deve ser ao acaso, mas feito de modo intencional para permitir a ancoragem (conexão) com os subsunçores específicos.

Os subsunçores, também denominados de conceitos subsunçores, são as estruturas cognitivas que o aprendiz já possui e que irão permitir a ligação dos novos conhecimentos. Em decorrência da ancoragem, os subsunçores podem ser modificados; portanto, pode ser entendido como um ciclo: o aprendiz recebe uma informação, esta é ancorada em um ou mais subsunçores, estes são modificados e outra vez recebem novas informações e reinicia-se, assim, o processo de cognição (SILVA, DEL PINO, 2019). Este pode ser definido como “dinâmico e o conhecimento vai sendo construído” (MOREIRA, 2010, p. 18). Este autor defende que, para que o processo de aprendizagem ocorra, deve-se conhecer a estrutura cognitiva pré-existente no aluno, composta por conceitos, ideias e proposições e suas respectivas clareza, estabilidade e diferenciação (MOREIRA; MASINI, 2006).

Quando uma nova informação é vinculada a aspectos relevantes e preexistentes na estrutura cognitiva e este processo modifica tanto a nova informação, quanto a estrutura preexistente, é denominada por Ausubel de Teoria da Assimilação. A assimilação da aprendizagem pode ser classificada em: significativa e mecânica (AUSUBEL; NOVAK; HANESIAN, 1983), apresentada graficamente na Figura 1.

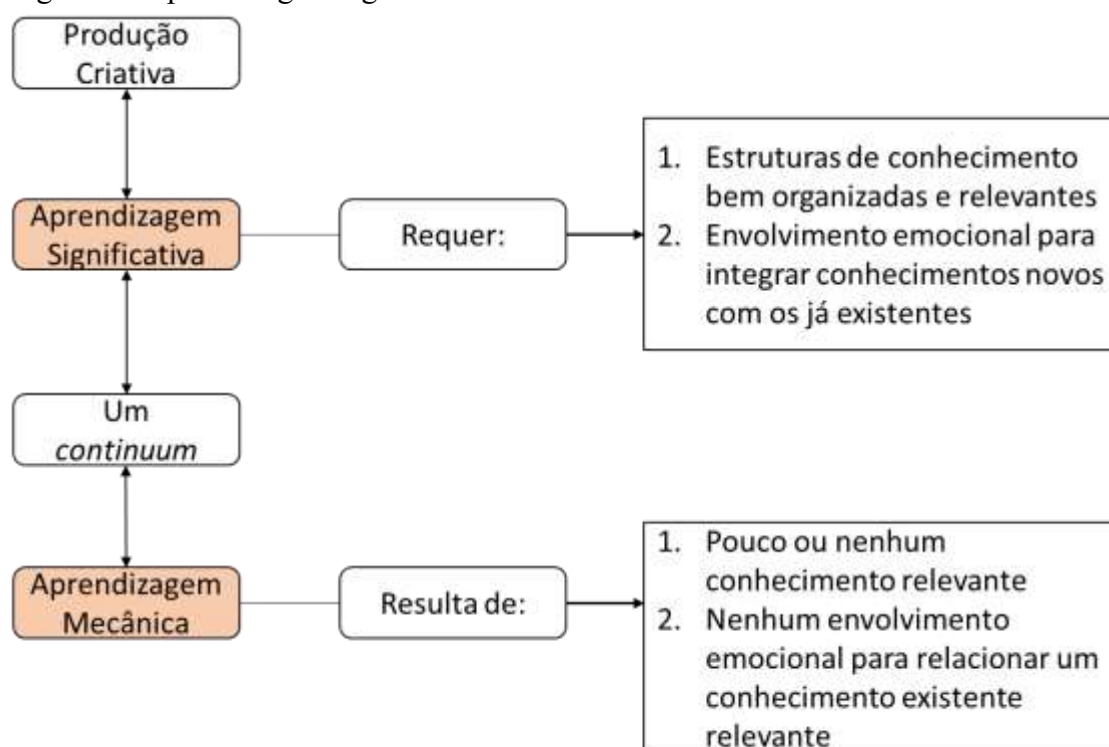
Figura 1 - Teoria de Ausubel



Fonte: Da autora (2020).

Conforme se pode acompanhar na Figura 1, apresentada anteriormente, a aprendizagem por assimilação mecânica acontece quando um novo conhecimento é apresentado ao aprendiz e este não encontra, por vários motivos, uma estrutura cognitiva para se conectar, sendo então armazenado (incorporado) de forma arbitrária e não substantiva. A aprendizagem, denominada por Ausubel (1968) como significativa, se dá quando uma ideia, conceito ou proposição são ancorados em conhecimentos pré-existentes. Estes dois tipos de processos não são antagônicos, mas sim geram um *continuum* entre a aprendizagem significativa e a mecânica, que pode ser representado conforme expressado na Figura 2.

Figura 2 - Aprendizagem significativa e mecânica



Fonte: Novak (2010, p. 24, tradução nossa).

A distinção entre as duas formas de aprendizagem é estabelecida pelo aprendiz, pois a assimilação dependerá da sua estrutura cognitiva existente, ou seja:

Quanto mais se relaciona o novo conteúdo de maneira substancial e não-arbitrária com algum aspecto da estrutura cognitiva prévia que lhe for relevante, mais próximo se está da aprendizagem significativa. Quanto menos se estabelece esse tipo de relação, mais próxima se está da aprendizagem mecânica (PELIZZARI *et al.*, 2002, p. 39).

Ainda na Figura 2, Novak (2010) explica que a criatividade é entendida como o mais alto nível da aprendizagem significativa.

Os tipos, as formas, as condições e as evidências da aprendizagem significativa estão descritas a seguir, nesta ordem.

Com relação aos tipos de aprendizagem significativa, Ausubel classifica em representacional, proposicional e conceitual. O primeiro – representacional – é o tipo básico do qual dependem os demais, consistente em atribuir significados aos símbolos (desconhecidos) ou o que representam, de forma individual (AUSUBEL, 2003; AUSUBEL; NOVAK; HANESIAN, 1983). Por exemplo, atribuir um som a uma figura.

A aprendizagem de proposições consiste em captar o significado de novas ideias expressadas em formato de proposição, não sendo simplesmente o somatório do conjunto de conceitos agrupados (AUSUBEL; NOVAK; HANESIAN, 1983). Em outras palavras: não significa saber o que palavras isoladas ou combinadas representam, mas sim aprender o significado de ideias (contexto) em forma de proposição. Por exemplo, o entendimento sobre algum aspecto social.

Já a aprendizagem significativa de Conceito é, além de ser considerada importante para a aquisição de conhecimentos, representada através de símbolos individuais e define que os conceitos são ideias unitárias ou genéricas. Este tipo de aprendizagem está relacionado com a representacional e pode ser entendido como que:

[...] os conceitos, assim como os objetos e as situações, se representam através de palavras ou nomes, apreender o que significam as *palavras conceptuais* (apreender qual o conceito representado por uma determinada palavra conceptual nova ou que a nova palavra conceptual possui um significado equivalente ao do próprio conceito) é, evidentemente, um importante tipo de *aprendizagem representacional*. Geralmente, segue a própria aprendizagem conceptual, visto ser muito conveniente conseguir representar-se os múltiplos atributos de critérios de um conceito recentemente apreendido com uma única palavra, que é equivalente ao mesmo em significado (AUSUBEL, 2003, p. 85).

Esse autor explica que muitas vezes os conceitos e o significados se entrelaçam, sendo assim equivalentes para sua assimilação.

Quanto à forma da aprendizagem significativa, do mesmo modo como foi delineada na teoria da assimilação, foi dividida em subordinada, superordenada (ou subordinante) e combinatória (AUSUBEL, 2003). A primeira, denominada de “subordinada” ou também de “subsunção”, faz a ancoragem de novos conceitos ou proposições em um subsunçor (ou vários) já existente, ou seja, considerando que a estrutura cognitiva está organizada de forma hierárquica, “a relação ocorre de subordinação do novo material a ideias mais subordinantes existentes na estrutura cognitiva” (AUSUBEL, 2003, p. 93). A subsunção pode ser derivativa ou comparativa. A Figura 3 apresenta graficamente esses tipos com uma explicação de cada uma das formas de aprendizagem.

Quando os novos significados modificam e diferenciam os subsunçores, este processo é denominado de Diferenciação Progressiva. Ou seja, quando um conceito mais geral é primeiramente introduzido, na sequência destes os conceitos mais detalhados ou mais específicos são abordados. Segundo Moreira e Masini (2006), este é o formato natural de como a consciência funciona quando um novo conhecimento é apresentado. A Diferenciação Progressiva está vinculada à forma de aprendizagem dita subordinada (o novo conceito fica subordinado a conceitos já existentes na estrutura cognitiva). Por exemplo, na modelagem Entidade Relacionamento de Banco de Dados, está representado como uma Especialização, na qual a entidade entidade-pai é Veículo e as entidades-filhas (ou subordinadas) são Carro e Moto.

A forma mais habitual da aprendizagem significativa, chamada de aprendizagem “superordenada” ou “subordinante”, acontece quando um conceito ou proposição mais geral é ancorado em estruturas subordinadas. Um exemplo elucidado que quando as crianças aprendem que os conceitos de “cenouras, ervilhas, feijões, beterrabas e espinafres se podem subordinar todos ao novo conceito subordinante vegetal” (AUSUBEL, 2003, p. 95).

Já o tipo “combinatória” acontece quando um novo conceito ou proposição não é possível subordiná-lo e tampouco ter subordinados, formando um variado conjunto de combinações de conhecimentos prévios já ancorados (MOREIRA, 2006, 2010). Ausubel (2003, p. 95) comenta que: “ao contrário das proposições subordinadas ou subordinantes, não são relacionais para com ideias relevantes particulares dentro da estrutura cognitiva”.

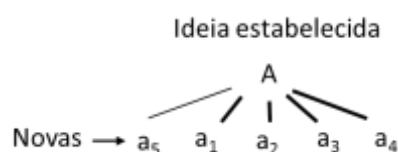
A Reconciliação Integrativa é representante da forma superordenada e combinatória. Ela explora os conceitos e ideias e suas relações, as diferenças e similaridades reconciliando as inconsistências, sejam elas reais ou aparentes, recombinao ou reorganizando, podendo inclusive corrigir preconceitos (NOVAK, 1983).

Figura 3- Formas de Aprendizagem Significativa

1. Aprendizagem Subordinada:

Subsunção Derivativa

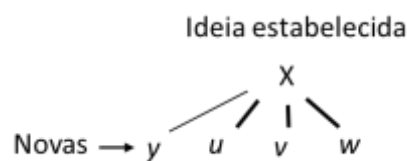
A nova informação a_5 está ligada à ideia subordinante A e representa outro caso ou extensão de A . Os atributos de critérios do conceito A não se encontram alterados, mas reconhecem-se os novos exemplos como relevantes.



Subsunção Correlativa

A nova informação y está ligada à ideia X , mas é uma extensão, alteração ou qualificação de X . Os atributos de critérios do conceito de

subsunção podem alargar-se ou alterar-se com a nova subsunção correlativa.



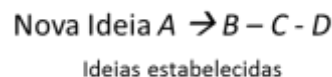
2. Aprendizagem Subordinante:

As ideias estabelecidas a_1 , a_2 , a_3 reconhecem-se como exemplos mais específicos da nova ideia A e tornam-se ligadas a A . A ideia subordinante A define-se através de um novo conjunto de atributos de critérios que acompanham as ideias subordinadas.



3. Aprendizagem Combinatória:

Considera-se que a nova ideia A está relacionada com as ideias existentes B , C e D , mas não é mais inclusiva nem mais específica do que as ideias B , C e D . Neste caso, considera-se que a nova ideia A tem alguns atributos de critérios em comum com as ideias preexistentes.



Fonte: Adaptado pela autora com base em Ausubel (2003, p. 111).

Com respeito aos requisitos ou condições para que a aprendizagem seja significativa (Figura 1), são elencadas a predisposição para aprender ou motivação, a presença de conhecimentos prévios e adequados (ou subsunçores) e os materiais utilizados, que sejam potencialmente significativos. O aluno deve manifestar uma atitude ou disposição para aprender de forma significativa, relacionando o novo conhecimento de forma não-arbitrária e substantiva (entendendo e relacionando as estruturas cognitivas) (AUSUBEL; NOVAK; HANESIAN, 1983).

Os materiais instrucionais devem estar organizados de modo lógico, estruturado e aprendível. O estudante deve apresentar conhecimentos prévios para interpretar os instrumentos apresentados. Assim sendo, é importante retomar que, para David Ausubel, a variável que mais influencia na aprendizagem são os conhecimentos prévios do aprendiz. Ou seja, a aprendizagem deve partir do que o aprendiz já sabe (AUSUBEL; NOVAK; HANESIAN, 1983).

A ancoragem está atrelada à intenção do aprendiz, aliada ao material potencialmente significativo, pois ambos devem estar presentes, já que um depende do outro (MOREIRA; MASINI, 2006). Nesta teoria, o importante é o relacionamento do novo conhecimento com algum subsunçor. Quando não existe o subsunçor (ou nenhuma forma de ancoragem) ao novo conhecimento, Ausubel sugere a utilização dos organizadores prévios (AUSUBEL; NOVAK; HANESIAN, 1983).

O uso de organizadores prévios é uma estratégia para “manipular a estrutura cognitiva” e deveriam ser apresentados antes do material a ser aprendido, ou seja, funcionam como material introdutório, de mais alto nível, com o objetivo de despertar no aprendiz a percepção do que ele sabe e o que precisa aprender. Estes organizadores prévios representam uma “ponte cognitiva” entre o conhecimento ancorado e o novo dado que está em contato, ou seja, sua relacionabilidade. Ainda, podem ser classificados em “explicativo” (usados para prover subsunçores relevantes), “comparativo” (integrar o novo conhecimento com um subsunçor similar) e “específico” (desenvolvido para a atividade a ser trabalhada). Os organizadores prévios são utilizados para facilitar a aprendizagem significativa de tópicos ou conteúdos mais específicos. Já para a aprendizagem significativa de vários tópicos, com o uso de materiais introdutórios, são qualificados como pseudo-organizadores prévios (AUSUBEL; NOVAK; HANESIAN, 1983). Como exemplos de organizadores prévios utilizados como estratégia de aprendizagem pode-se mencionar as apresentações, discussões, textos, mapas conceituais, figuras, programas, entre outros.

A coleta das evidências, que demonstra a ocorrência da aprendizagem significativa, não é uma tarefa simples, pois “a compreensão genuína implica a posse de significados claros, precisos, diferenciados e transferíveis” (AUSUBEL, 2003, p. 130). Para este estudioso, os testes propostos aos aprendizes devem avaliar a compreensão por intermédio de diferentes linguagens, ou contextos, ou ainda, do material instrucional já exposto. O aprendiz, para resolver uma questão ou problema, deve utilizar os subsunçores, e não apenas repetir os conceitos, quer dizer, deve ser capaz de aplicar e transformar o conhecimento existente.

O mesmo autor elucida que:

[...] a resolução de problemas é um método válido e prático de se avaliar a compreensão significativa de ideias. Contudo, não é o mesmo que dizer que o aprendiz que não consegue resolver um conjunto representativo de problemas, com base num determinado grupo de material de instrução, não compreende necessariamente, mas tem apenas memorizados os princípios exemplificados por estes problemas. A resolução de problemas bem sucedida exige muitas outras capacidades e qualidades – tais como poder de raciocínio, perseverança, flexibilidade, ousadia, improviso, sensibilidade aos problemas e astúcia táctica – além da compreensão dos princípios subjacentes (AUSUBEL, 2003, p. 130).

Como bem exposto pelo autor, a resolução de problemas impõe uma série de habilidades inerentes ao processo, bem como de forma isolada não pode ser utilizada como determinante da aprendizagem.

Além da resolução de problemas, outra técnica de avaliar a aprendizagem significativa é a sugestão de tarefas a serem desenvolvidas de forma que estejam interligadas, sendo cada uma dependente da realização e domínio da anterior (MOREIRA; MASINI, 2006).

Há outros pontos a serem considerados quanto ao desempenho da aprendizagem do aluno (AUSUBEL, 2003). Por exemplo: quando o aprendiz resolve um problema pelo método de tentativa e erros, ou por tentativas aleatórias, ou compreende a classe de problema e mecanicamente o resolve. Para evitar isso, pode-se sugerir ao professor a elaboração e proposição de questões diferentes daquelas que os estudantes estão acostumados a resolver, propondo situações novas e não familiares, exigindo do aprendiz a máxima transformação do conhecimento existente.

A partir do que Ausubel (2003) sustenta sobre as evidências da aprendizagem significativa, torna-se claro e coerente usar esta teoria nesta tese, pois ao tratar sobre PC busca-se a resolução de um problema, por meio da representação algorítmica e/ou linguagem de programação. Para tal, o aprendiz precisa dominar o conceito (entendimento do significado dos comandos de programação) e sua proposição (no caso de programação, seu uso para resolver o problema). Uma vez que um problema seja resolvido, ou, mais especificamente, tornou-se um programa ou *software*, e, se funciona adequadamente, então está terminado. Certamente, parte-se para a resolução de outros problemas, com outras necessidades ou requisitos, a fim de resolvê-los, buscando-se o conhecimento anterior – ou seja, os subsunçores. Não há motivo para que o aprendiz resolva e desenvolva o mesmo programa, seria incoerente e não significativo. Com base no desenvolvimento das habilidades de PC, é importante entender o que são conceitos, os quais são apresentados a seguir.

2.3.1.1 Conceitos

Os conceitos são a chave para o desenvolvimento cognitivo, pois eles representam a compreensão de novos conhecimentos e do uso correto e significativo na resolução de problemas, segundo Ausubel (2003).

Segundo esse mesmo autor, há duas formas de aprender os conceitos. A primeira, denominada de formação de conceitos é típica da criança em idade pré-escolar. Por não corresponder ao escopo deste trabalho, não será detalhada. Em contrapartida, a segunda forma é a assimilação de conceitos, típica de adultos (e crianças em idade escolar), na qual o processo de aprendizagem acontece por “ancoragem em conceitos já existentes na estrutura cognitiva do aprendiz” (MOREIRA, 2010, p. 74). Esta ancoragem é resultante de uma situação consciente, diferente e idiossincrática de cada aprendiz e é dependente dos seus subsunçores, do nível de desenvolvimento do aprendiz, da sua habilidade intelectual e do modo como é exibido o material

instrucional (MOREIRA; MASINI, 2006). Importante destacar que a assimilação ausubeliana é um processo ativo do ponto de vista cognitivo que acontece no aprendiz.

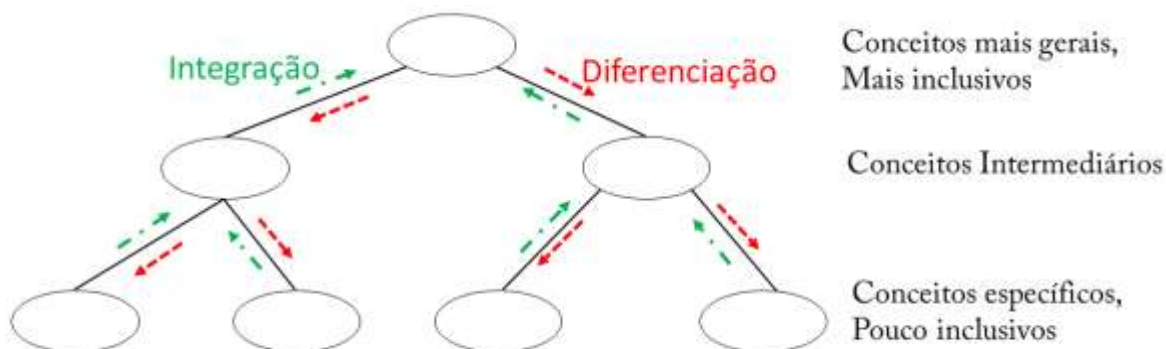
Igualmente, a linguagem exerce um papel fundamental no processo de aprendizagem, pois facilita a comunicação entre os participantes da interação. Os conceitos apreendidos podem ser utilizados de diferentes formas:

1. Aquisição de novos conceitos;
2. Categorização perceptual da experiência;
3. Solução de problemas;
4. Percepção de novos significados dos conceitos e proposições previamente aprendidos (MOREIRA; MASINI, 2006, p. 40).

Cabe salientar que a aquisição de conceitos pode se entrelaçar com a solução de problemas, porém nem sempre uma depende da outra. Ou seja, pode existir a solução de problemas sem o uso de conceitos ou vice-versa.

Em seu livro “Uma teoria de Educação”, Novak (1981), incorpora à teoria da aprendizagem significativa o entendimento de que o aprendizado é o conjunto de experiências (cognitivas, afetivas e psicomotoras) que contribuem para o engrandecimento (*empowerment*) do indivíduo, com a finalidade de atender sua vida diária. Essa teoria apresenta, como premissa fundamental, que os seres humanos pensam, sentem e atuam, de forma positiva ou negativa, e sempre integrados. Portanto, seja qual for o evento educativo, segundo este autor, uma ação pode trocar significados (pensar) e sentimentos entre o aprendiz e o professor. A partir das estruturas cognitivas hierárquicas, o ensino deve estar organizado, “descendo e subindo”, apresentando os conceitos mais gerais, a seguir os mais específicos e voltando a integrá-los com mais genéricos, conforme pode ser visualizado na Figura 4.

Figura 4 - Representação esquemática do Modelo Ausubeliano



Fonte: Adaptado pela autora com base em Moreira e Masini (2006, p. 33).

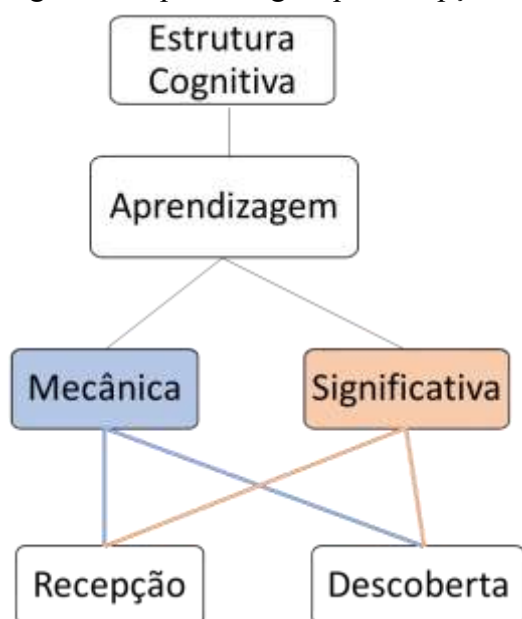
Pode-se perceber, na Figura 4, que a partir da apresentação de um conceito este pode ser realizado de forma diferenciada e/ou integrada, dependendo da estrutura cognitiva do aprendente, sendo um processo bastante particular.

Com efeito, sobre as premissas citadas, para o sujeito adulto deste trabalho, o pensar pode ser entendido como o movimento dos subsunçores empregados na solução de um problema (por exemplo, profissional), após perceber a demanda ou necessidade e atuando na sua solução, por exemplo, propondo um programa.

Na tese de Rehfeldt (2009, p. 31), a autora apontou alguns conhecimentos prévios que os sujeitos da pesquisa, alunos do Curso de Administração, que eram “necessários para iniciar a modelagem de situações-problema de programação linear”, porém concluiu que os aprendentes não o possuíam. Este exemplo reforça a importância do professor/tutor reconhecer o que o aprendiz já sabe e planejar os organizadores prévios de acordo com a situação; em outras palavras: “para os professores é um grande desafio ensinar levando em conta estas duas variáveis: o conhecimento prévio e a intencionalidade de aprender do aluno” (MOREIRA, 2013, p. 1).

Ainda, Ausubel, Novak e Hanesian (1983) apontam que tanto a aprendizagem significativa quanto a mecânica podem ser quaisquer dos tipos básicos de aprendizagem: por recepção e por descoberta (Figura 5).

Figura 5 - Aprendizagem por recepção e descoberta



Fonte: Da autora (2020).

A aprendizagem por recepção significativa é também denominada como aprendizagem verbal significativa receptiva, porque esta teoria é classificada como significativa e cognitivista, que significa “aquela que resulta do armazenamento organizado de informações na mente do ser que aprende, e esse complexo organizado é conhecido como estrutura cognitivista” (MOREIRA; MASINI, 2006, p. 95). Também é verbal, pois seu autor considera a linguagem como o mais importante instrumento para a aprendizagem significativa, cumprindo um papel operacional; sendo “o principal meio de adquirir conhecimento” (AUSUBEL; NOVAK; HANESIAN, 1983, p. 37) e complementado por Moreira e Masini (2006, p. 77): “isso vale principalmente para a adolescência e a idade adulta”. Assim como é receptiva, porque este “é o mecanismo humano, por excelência, para adquirir e armazenar a vasta qualidade de ideias e informações de qualquer campo de conhecimento” (AUSUBEL, 2003, p. 81). Na aprendizagem por recepção (seja ela mecânica, seja significativa), o conteúdo é apresentado ao aprendiz na sua forma final. Ao estudante compete a internalização do conhecimento e a disposição para aprender, não envolve nenhuma descoberta, mas também não é passivo. Somando-se a isto, o material instrucional a ser utilizado deve ser potencialmente significativo. Quer dizer, relacioná-lo de forma não arbitrária com ideias relevantes, especificamente (AUSUBEL; NOVAK; HANESIAN, 1983). Importante ressaltar a palavra ‘potencialmente’, pois se o material já fosse significativo, não se necessitaria da aprendizagem.

Já na aprendizagem por descoberta (Figura 5), a formação de conceitos ou resolução de problemas não são apresentadas pelo professor, e sim devem ser descobertos, ou pesquisados pelo sujeito aprendente.

Os mesmos autores também explicam que na aprendizagem por descoberta (mecânica ou significativa) o aprendiz deve reagrupar as informações, associando-as à estrutura cognitiva, a fim de construir o produto final desejado que represente a solução para o problema ou caminhos para a sua resolução. No final, o conteúdo ou conhecimento descoberto torna-se significativo, igualmente como acontece na aprendizagem por recepção. Para determinados conteúdos ou propósitos específicos, a aprendizagem por descoberta é recomendada. A saber, especialmente adotado para uma aprendizagem de método científico (utilizado para realizar descobertas de novos conhecimentos). Porém, há de se observar que este não é um método viável para ensinar conteúdos muito extensos que se referem a disciplinas de estudos (AUSUBEL; NOVAK; HANESIAN, 1983).

São formas de aprendizagem significativa por descoberta a resolução de problemas e a criatividade. Ambas as formas são diretamente ligadas ao desenvolvimento do pensamento computacional (PC), que muitas vezes é descrito como um método de resolver problemas. A

definição de David Ausubel sobre a resolução de problemas é “qualquer atividade que tanto a representação cognitiva da experiência prévia, como os componentes de uma situação problemática apresentada, são reorganizados para alcançar o objetivo” (AUSUBEL; NOVAK; HANESIAN, 1983, p. 486, tradução nossa). Para resolver o problema, o aprendiz precisa compreender as condições elencadas e assimilar a sua solução, sendo, neste momento, o uso da aprendizagem significativa por descoberta.

O produto da resolução significativa do problema refere-se às características finais da solução do problema, podendo gerar no indivíduo um “sentimento de prazer da correta descoberta de ‘ver a luz’ ou de dizer eureka!” (AUSUBEL; NOVAK; HANESIAN, 1983, p. 488, tradução nossa). Esse sentimento é muito percebido em aulas de programação, quando o aluno consegue resolver ou implementar um programa sem erro.

Retoma-se aqui um ponto importante da teoria ausubeliana (AUSUBEL; NOVAK; HANESIAN, 1983), isto é, a ênfase de que a aprendizagem por recepção, se bem verbalizada, com materiais instrucionais significativos, exige do aprendiz um movimento cognitivo de ancoragem dos novos conceitos em sua estrutura cognitiva.

Descreve-se, na sequência, a teoria proposta por Knowles sobre aprendizagem de adultos.

2.3.2 Andragogia

A palavra Andragogia é formada a partir dos vocábulos gregos *andros* (homem) + *agein* (conduzir) + *logos* (tratado, ciência) (KNOWLES, 1973). Fazendo referência à ciência da educação de adultos, a **Andragogia** faz oposição à Pedagogia, proveniente igualmente dos termos gregos *paidós* (criança) + *agein* (conduzir) + *logos* (tratado ou ciência), referindo-se à ciência da educação de crianças.

Em 1981, Knowles (1981, p. 43), definiu a andragogia “como a arte e a ciência de ajudar os adultos a aprender, em contraste com a pedagogia como a arte e a ciência de ensinar as crianças”. A Andragogia deve ser entendida como a filosofia, a ciência e a técnica da educação de adultos (CAVALCANTI; GAYO, 2005).

Estes pesquisadores ainda complementam que os adultos e as crianças aprendem de formas diferentes. Em 2005, Knowles, Holton e Swanson (2005) descrevem como sendo um

modelo de pressupostos sobre os alunos a serem usados juntamente com o modelo de postulados¹⁹.

Em 1970, Malcolm Knowles propõe um conjunto de princípios sobre a aprendizagem de adultos e passa ser considerado o pai da andragogia nos Estados Unidos, defendendo que “não se pode ensinar um adulto, mas, sim, ajudá-lo a aprender” (KNOWLES; HOLTON; SWAMSON, 2011, p. 13). Portanto, o ponto essencial da Andragogia repousa em um conjunto de princípios de aprendizagem de adultos, aplicáveis a qualquer situação.

Ensinar adultos é diferente de ensinar crianças. O professor²⁰ andragógico deve considerar que o processo de aprendizagem está centrado no aluno, que participa da elaboração do programa (conteúdos) a ser trabalhado, percebendo a aplicação prática. Considerando que o aprendente já possui experiências (CAVALCANTI; GAYO, 2005) e, a partir do desenvolvimento de atividades, baseado em respeito e confiança, o professor passa a ser um tutor ou facilitador para os adultos. Para Alcalá (2010), o facilitador orienta a aprendizagem e deve estar muito bem preparado para este papel, sendo responsável pela andamento e execução da práxis andragógica.

Para Cavalcanti e Gaio (2005, p. 49), a avaliação andragógica é:

[...] contínua, constante, diagnóstica. Visa, a cada momento, detectar falhas (não compreensão de conceitos, aprofundamento insuficiente do raciocínio dedutivo ou indutivo na discussão de problemas, falhas no interesse e participação, entre outros) de modo que sejam prontamente corrigidas.

As falhas devem ser verificadas no decorrer do processo, pois no futuro profissional, através da sua própria avaliação, precisa perceber quando seu desempenho não está a contento.

Eis que surge um questionamento a respeito de quando um indivíduo se torna adulto. Existem conceitos diferentes sobre o desenvolvimento da pessoa humana, suas capacidades e seu amadurecimento. Sob o ponto de vista sociológico, a pessoa se torna um adulto quando assume um papel produtivo na sociedade, como o de trabalhador, ou votante, também associada ao

¹⁹ Em 1977, Félix Adam cunhou o termo *Antropogogia* (escrita em espanhol) e definiu como sendo “es la ciencia y arte de instruir y educar permanentemente al hombre en cualquier periodo de su desarrollo psicobiológico en función de su vida cultural, ergológica y social” (CASTILLO, 2018a). Com o passar do tempo, essa definição foi modificada para *Ciencias Agógicas* (em espanhol) que compreende: “La *Paidagogía*, estudia la educación de niños en su etapa de preescolar de 3 hasta 6 años de edad. La *Pedagogía* estudia la educación del niño en su etapa de Educación Básica. La *Hebegogía* estudia la educación del adolescente en su etapa de Educación Media y Diversificada (12 a 18 años). La *Andragogía* estudia la educación de las personas adultas hasta la madurez. La *Gerontogía* estudia la educación de adultos mayores” (YTURRALDE, 2018, texto digital).

Os autores Hase e Kenyon (2000), definem o termo *Heutagogia* (do inglês *Heutagogy*) como sendo a aprendizagem autodirigida. É uma extensão da Andragogia no sentido de focar o processo de ensino e aprendizagem frente a mudanças tecnológicas. Pode ser entendida como uma abordagem para o século XXI, bem como para o ensino a distância (COELHO; DUTRA; MARIELI, 2016).

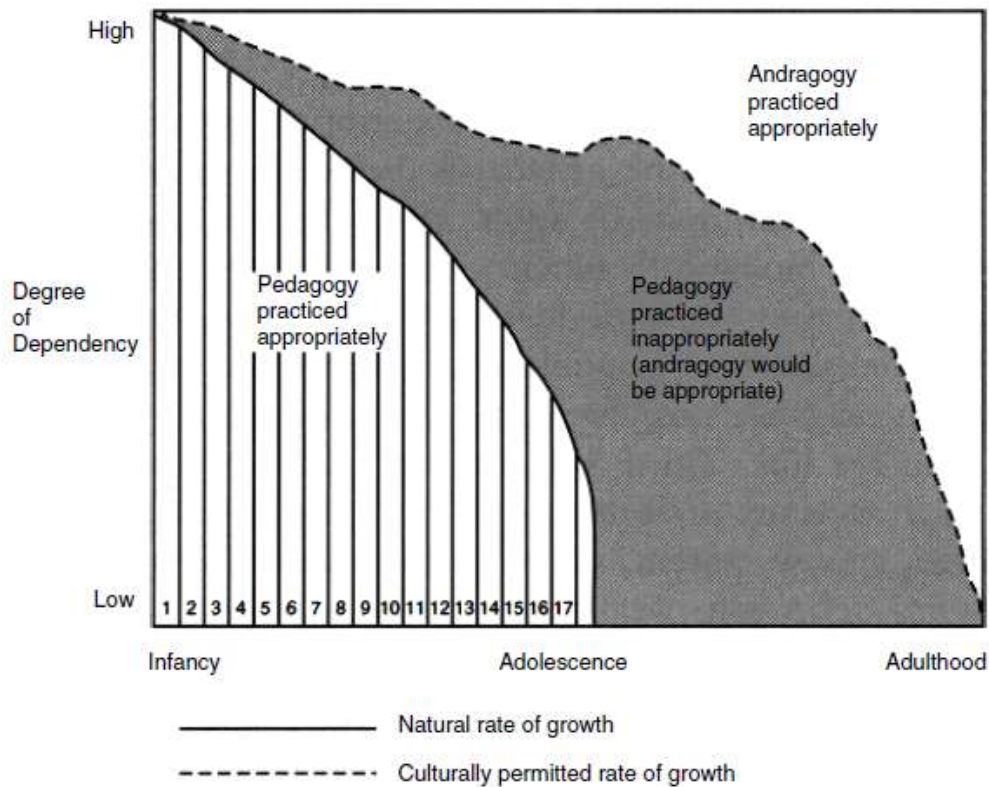
²⁰ No contexto da Andragogia, adotado também neste trabalho, professor e facilitador são sinônimos, bem como aluno e aprendiz.

aspecto econômico. Segundo o ponto de vista biológico, o indivíduo se torna adulto quando é capaz de reproduzir-se. Ou ainda, quando a pessoa é considerada apta a responder por seus atos na convivência social e nas normas legais, quando atinge a chamada maturidade jurídica. Por último, na definição sob o ponto de vista psicológico, o qual está ligado à independência psíquica, a pessoa se torna responsável por sua própria vida, conquistando sua competência autoadministrativa (CARVALHO *et al.*, 2010; CAVALCANTI, 1999).

A fase adulta traz a maturidade e a independência, tornando os indivíduos capazes de analisar, criticar, avaliar situações, tomando em conta suas experiências. Nesta fase, o importante é experienciar e vivenciar os momentos para que se tornem significativos, ou seja, o aprender fazendo. Bellan (2018) e Cavalcanti (1999) apontam pesquisas que demonstram que os adultos armazenam apenas 10% do que ouvem após decorridas 72 horas. Complementando este dado, “Kelvin Miller afirma que estudantes adultos [...] serão capazes de lembrar de 85% do que ouvem, vêem e fazem, após o mesmo prazo” (CAVALCANTI, 1999, p. 1) e que são os primeiros 15 minutos do encontro (seja aula, seja palestra) nos quais os dados passados são mais absorvidos.

Com o crescimento, o indivíduo desenvolve suas habilidades, esperando uma aplicação prática do conhecimento, resolvendo problemas reais ligados ao seu cotidiano ou com possíveis melhorias em sua vida (KNOWLES; HOLTON; SWAMSON, 2011). Passa a ser autodirigido, utilizando seus conhecimentos e sua própria capacidade de aprender. A Figura 6 mostra as fases do crescimento.

Figura 6 - Maturação natural x Grau de dependência



Fonte: Knowles (2005, p. 63).

Na Figura anterior, pode-se perceber a relação da idade cronológica e a teoria apropriada para ser utilizada. No intervalo entre a adolescência e a vida adulta (aqui considerando qualquer um dos entendimentos quando inicia a idade adulta, já apresentados), o autor sugere que os princípios da andragogia já comecem a ser utilizados. Nesse hiato, pode-se preconizar que ambas sejam consideradas, não como excludentes, mas sim como complementares.

A Figura 7, de forma gráfica, mostra a andragogia descrita por Knowles como sendo um conjunto de princípios de aprendizagem de adultos. Tais princípios estão divididos em três dimensões, da mais externa para a mais interna: objetivos e propósitos para a aprendizagem, diferenças individuais e situacionais, e andragogia: princípios fundamentais da aprendizagem de adultos.

Figura 7 - Modelo da andragogia de Knowles



Fonte: Knowles, Holton e Swamson (2011, p. 21).

A dimensão mais externa, conforme disposta na figura anterior (Figura 7), apresenta os objetivos e propósitos da aprendizagem e representa os resultados do desenvolvimento individual. Dividida em crescimento: individual (aperfeiçoamento pessoal, com foco no aprendiz), institucional (aumento da produtividade) e social (transformação social). Argumentam Knowles e seus colegas que “a andragogia é uma estrutura de aprendizagem individual, mas pode decorrer do objetivo de crescimento individual, institucional ou social” (KNOWLES; HOLTON; SWAMSON, 2011, p. 144). Em outras palavras, tudo o que uma pessoa obtenha no seu desenvolvimento beneficiará tanto sua instituição, quanto seu entorno (CASTILLO, 2018b).

A dimensão mediana – Diferenças individuais e situacionais – são caracterizadas como variáveis e se agrupam em diferenças de tema, de situação e de caráter individual. Quanto às diferenças de tema, que são altamente variadas e de contexto diferentes, exigem distintas abordagens de aprendizagem, ainda que os conteúdos não sejam de conhecimento do aluno.

Quando a situação produz um efeito na aprendizagem, considera-se uma diferença situacional, como, por exemplo, o acesso à internet. Além disso, o contexto social ou cultural

também pode contribuir neste processo. Por último, as diferenças individuais, que podem ser cognitivas de personalidade ou conhecimento, e a experiência do aprendiz, sua estratégia de aprendizagem, suas habilidades, entre outros fatores, influenciam e/ou motivam o aluno para aprender.

Já, no centro do modelo, aparecem os seis princípios da andragogia, os quais devem ser considerados na prática (KNOWLES; HOLTON; SWAMSON, 2011). Todos estão listados, a seguir, bem como sua interpretação correspondente:

(1) a necessidade do aprendiz de saber: os adultos precisam saber o porquê de aprender, qual o propósito, como e onde poderá ser aplicado este conhecimento e como será a aprendizagem. Neste princípio, o papel do professor-facilitador é importante, no sentido de despertar o aprendiz para novos horizontes do conhecimento;

(2) o autoconceito do aprendiz: o adulto parte da premissa de que é responsável por suas decisões; portanto, sente-se capaz de autodirigir-se. O educador de adulto deve ajudar o aprendiz, nessa situação, porque ao longo da vida escolar os alunos se colocam como aprendizes dependentes do professor, isto é, esperando que este “o ensine”;

(3) a experiência anterior do aprendiz: em uma atividade educacional com adultos, há que se considerar a “heterogeneidade de formação, estilo de aprendizagem, motivação, necessidades, interesses e objetivos” (KNOWLES; HOLTON; SWAMSON, 2011, p. 73). Portanto, valorizar suas experiências é primordial, pois se isso não ocorre os indivíduos adultos se sentem rejeitados tanto no sentido de conhecimento, quanto pessoal. Para enfatizar, julga-se interessante expor as próprias palavras dos autores:

Logo, a ênfase da educação de adultos está nas técnicas experienciais - técnicas que utilizam a experiência dos aprendizes como discussões em grupo, exercícios de simulação, atividades de resolução de problemas, estudos de caso e métodos de laboratório, em vez de técnicas de transmissão. Também há uma ênfase maior nas atividades de ajuda aos colegas (KNOWLES; HOLTON; SWAMSON, 2011, p. 73).

(4) a prontidão para aprender: os adultos tornam-se receptivos à aprendizagem quando esta se relaciona às situações reais, sincronizando os seus conhecimentos com as atividades de desenvolvimento e com a vontade de aprender. O professor pode usar estratégias para induzir o desejo da aprendizagem, como, por exemplo, em exercícios de simulação;

(5) a orientação para a aprendizagem: a aprendizagem do adulto está centrada no problema ou na tarefa a ser resolvida, ou seja, que tenha uma conexão com sua vida. Essa é a motivação para empenhar-se, assimilando novos conhecimentos e habilidades. O professor deve ser sensível e conhecer o aprendiz e seu entorno. Boa parte dos alunos adultos são de cursos

noturnos e passam o dia no trabalho, por isso é importante a objetividade e, na medida do possível, fazer a ponte com seu cotidiano;

(6) a motivação para aprender: para Knowles, Holton e Swanson (2011), a motivação pode ser externa e interna. Melhor emprego, maiores salários, promoções, entre outros, são exemplos de motivação externa. Como exemplo de motivação interna do indivíduo pode-se destacar a melhoria da autoestima ou da qualidade de vida. *A priori*, o adulto se motiva para melhorar e adquirir mais conhecimento, porém, muitas vezes, encontra dificuldades, como falta de recursos e/ou tempo disponível, bem como cursos não adequados aos princípios da formação de adultos. Paz (2017, p. 5) apresenta a experiência de um curso sobre novas tecnologias, baseado no PC, no nível de formação de professores, no qual descreve:

Ao planejar o conteúdo do curso, este precisa estar relacionado às atividades profissionais dos alunos, auxiliando na solução dos problemas encontrados em sua rotina, prevendo as diferenças de estilo de vida, ritmo de aprendizagem; deve ser incentivado o compartilhamento de experiências e conhecimentos através de situações interativas (discussões, debates, atividades em grupo) e durante as aulas deve ser estabelecido um relacionamento acolhedor, respeitoso, sem pressões ou competitividade.

Em suma, nesta citação, a autora corrobora os princípios andragógicos na prática com um grupo de 21 professores. Os momentos de aprendizagem sugeridos devem ser cooperativos entre professor e os alunos adultos desde o planejamento, bem como no desenvolvimento e na avaliação do processo de aprendizagem. Esse modo deve ser dinâmico, podendo ser revisto e alterado a qualquer tempo. O adulto sabe o que precisa aprender, geralmente busca um conhecimento prático, aplicável a curto prazo. Porém, cabe ao professor avaliar as necessidades (por exemplo, conteúdos que devem ser aprendidos) e orientar de forma criativa sua “aula”.

Knowles, Holton e Swanson (2011, p. 277) já comentavam sobre o uso da tecnologia, como “potencial utilização para a aprendizagem de adultos”. Este fato pode ser confirmado percebendo o fenômeno do uso dos computadores pessoais e da internet, mas também com a expansão do uso de dispositivos móveis (como *tablets* e *smartphones*). Nesse sentido, a diminuição do preço desses aparelhos permitiu que grande parte da população obtivesse acesso, bem como a mudança de comportamento que aconteceu com a oferta de cursos a distância, por exemplo, nos quais o aprendiz escolhe o que quer aprender, quando quer, ou seja, no momento mais adequado, e onde quer aprender, graças à ubiquidade proporcionada pelas tecnologias. Como consequência, o adulto, autodirigido, a partir de suas experiências, é capaz de buscar caminhos para sua aprendizagem, ajustando às suas problemáticas reais.

Em se tratando de prática na aprendizagem andragógica, Knowles (1981) propõe o uso do modelo de aprendizagem “todo-parte-todo”, de autoria de Swanson e Law²¹. As características relevantes para este trabalho estão explicitadas no Quadro 2, no final deste capítulo.

O ambiente no qual ocorre a aprendizagem deve ser horizontal entre o professor e os aprendizes, baseado em participação, respeito, responsabilidades e confiança, no qual os êxitos e fracassos são compartilhados. Ao professor cabe o entendimento de que seu aluno é um adulto ao avaliá-lo (KNOWLES, 1973) e que, segundo o modelo andragógico, o que deve acontecer é uma autoavaliação. Nesta, o aluno tem consciência do que sabe e aonde quer chegar para desenvolver seus conhecimentos e competências (KNOWLES; HOLTON; SWANSON, 2005). Com o objetivo de avaliar e medir a assimilação dos princípios, conteúdos e técnicas durante uma intervenção (curso, por exemplo), deve-se aplicar pré-testes e pós-testes, a fim de mensurar os ganhos específicos resultantes das experiências de aprendizagem. Knowles, Holton e Swanson (2011) enfatizam que, durante um processo de aprendizagem, deve-se diagnosticar as novas necessidades de aprendizagem, ou seja, a educação continuada do adulto, ou educação para a vida, e os resultados apresentados de forma qualitativa. Esses autores sugerem, ainda, que seja feito um acordo entre o professor e os alunos, no sentido de que todos os envolvidos no processo tenham pleno conhecimento sobre os papéis e responsabilidades de cada sujeito no processo de ensino e de aprendizagem. Logo, sugerem os chamados contratos de aprendizagens, apresentados a seguir.

2.3.2.1 Contratos de aprendizagem – CAPre

Os contratos de aprendizagem (CAPre), como o próprio nome sugere, são acordos firmados entre aprendizes, professores, tutores, colegas e instituições de ensino sobre o planejamento das experiências de aprendizagem, objetivos, estratégias e avaliação de aprendizagem. São importantes, porque envolvem o aprendiz (adulto e autodirigido) em todo o processo, tornando-o partícipe e responsável da experiência da aprendizagem, desenvolvendo no aluno o senso de domínio, sentimento de propriedade e compromisso (CAVALCANTI; GAYO, 2005; KNOWLES; HOLTON; SWAMSON, 2011).

²¹ SWANSON, Richard A.; LAW, Bryan D. **Performance Improvement Quarterly**, v. 6, n. 1, p. 43-55, 1993. Esse artigo foi apresentado pela primeira vez na *European Conference on Educational Research*, Enschede, Holanda, em 24 de junho de 1992.

Para Knowles, Holton e Swanson (2005), o uso dos CAPre resolve alguns problemas, tais como: diferentes interesses, motivações e habilidades dos alunos para aprender; estimula o senso de propriedade dos objetivos da aprendizagem e avaliações, entre outros.

Diferente dos métodos tradicionais, os CAPre, além de atribuírem um papel ativo ao indivíduo, podem ser recursos utilizados para melhorias, para inovação ou a gestão (CAVALCANTI, 1999). O primeiro passo para pensar sobre o CAPre são as necessidades de aprendizagem, levantadas através de uma avaliação pessoal, ou por lacunas de conhecimento, entre outras. A fim de projetar um CAPre deve-se ter clareza de onde o indivíduo se encontra e a lacuna de conhecimentos até seu objetivo, a saber, aonde quer chegar.

Um exemplo de um típico CAPre é mostrado na Figura 8.

Figura 8 - Contrato de Aprendizagem

Nome:			
Atividade:			
Objetivos de Aprendizagem	Recursos e Estratégias de Aprendizagem	Evidências de Realização dos Objetivos	Critérios e Meios para Validar Evidências

Fonte: Adaptado pela autora com base em Knowles, Holton e Swanson (2011).

No início do documento, mostrado na Figura 8, o indivíduo se identifica, assim como as atividades ou o nome do próprio curso é escrito. Elencadas as lacunas e/ou necessidades, cada uma se torna um item em “Objetivos de Aprendizagem” (veja-se a primeira coluna). Para cada objetivo, uma descrição de como este será realizado ou alcançado deve ser preenchido na coluna “Recursos e Estratégias de Aprendizagem”. Os recursos podem ser humanos, materiais, tecnológicos, e as estratégias, que incluem as ferramentas e técnicas que serão utilizadas. O próximo passo é preencher a coluna “Evidências de Realização dos Objetivos”, na qual será descrito o grau de evidência alcançado por cada objetivo. Por exemplo, para o objetivo Compreensão, uma evidência pode ser o conhecimento para resolver um problema.

E, por último, para cada objetivo descrito, definem-se os “Critérios e Meios para Validar Evidências” (coluna 4). Os critérios serão avaliados de acordo com o objetivo e a evidência, especificando os meios de seu julgamento. Como exemplos de critérios, pode-se elencar velocidade, precisão e criatividade; assim como os meios de avaliação que podem ser escalas, relatórios, entre outros (KNOWLES; HOLTON; SWAMSON, 2011). Este modelo de CAPre (Figura 8) foi o referencial utilizado neste trabalho.

Em suma, a educação de adultos é um “processo por meio do qual os aprendizes se tornam conscientes da experiência significativa” (KNOWLES; HOLTON; SWAMSON, 2011, p. 51), complementando com a teoria de Ausubel, a qual é nomeada de aprendizagem significativa. Esses autores convergem na ideia de que a aprendizagem deve ter um sentido para o aprendiz, e nesta seção o enfoque foi na aprendizagem significativa de adultos. Ainda, esses estudiosos enfatizam as atividades de resolução de problemas e criatividade (conceitos importantes para a prática do PC), conectado a vida real, com problemas cotidianos, principalmente na Andragogia.

Segundo Ausubel (2003), a maior incidência de aprendizagem significativa é do tipo subordinada, ou seja, a nova ideia aprendida encontra-se hierarquicamente subordinada a uma ideia preexistente. Formam-se os subsunçores que logo contribuem com a experiência do indivíduo. Esta experiência é a fonte mais rica para a aprendizagem dos adultos. Então, tem-se um complemento de ambas as teorias.

Quando Knowles, Holton e Swamson (2011) afirmam que os adultos são autodirigidos, significa que escolhem o que aprender, de acordo com suas necessidades, podendo fazer uma aprendizagem por descoberta, ademais que aumenta a independência conforme o avanço da idade.

No Quadro 2, um resumo comparativo das ideias propostas por ambas as teorias, apresentando suas equivalências.

Quadro 2 - Equivalências das Teorias Ausubeliana e de Adultos

Aprendizagem significativa	Aprendizagem de adultos
Fator principal é o que o aluno já sabe	Princípio número 3: experiência anterior do aprendiz
Subsunçor	Diferenças individuais: que podem ser cognitivas de personalidade ou o conhecimento e a experiência prévia do aprendiz
Assimilação	Tempo de aprendizagem /concentração do adulto
Assimilação significativa	Objetivos e propósitos para a aprendizagem
Aprendizagem por recepção	Princípio número 5: orientação para a aprendizagem
Aprendizagem por descoberta	Princípio número 2: autoconceito do aprendiz (autônomo e autodirigido)

Aprendizagem significativa	Aprendizagem de adultos
Material potencialmente significativo	Princípio número 1: necessidade de o aprendiz saber: o porquê, o que e como
Motivação para aprender	Princípio número 6: motivação para aprender
Avaliações - proposta de resolução de problemas diferentes daqueles já conhecidos a fim de exigir “máxima transformação do conhecimento existente” (MOREIRA; MASINI, 2006, p. 24); - tarefas interligadas	- Resoluções de problemas da vida do aprendiz
- Organizadores prévios - Motivação para aprender	Modelo todo-parte-todo (KNOWLES; HOLTON; SWAMSON, 2011): é um método sistemático de ensino: O primeiro “ TODO ”: duas funções principais: - oferecer uma estrutura mental, por meio de organizadores prévios e do alinhamento de <i>schemata</i> ²² , que preparam os aprendizes para o novo conteúdo a ser aprendido; - é “motivar o participante a querer aprender, tornando o conteúdo significativo e conectado a ele” (p. 233). - A motivação do aprendiz é um aspecto importante do modelo de aprendizagem todo-parte-todo, pois, “sem aprendizes que valorizem o novo conteúdo que estiver sendo ensinado, haverá pouca esperança de retenção do que se aprende e de transferência do que se aprende para o local de trabalho” (p. 234).
- Subsunçores que recebem nova informação que é incorporado ao subsunçor modificando-o, tornando um subsunçor diferente. - Diferenciação progressiva (Figura 4)	“ PARTE ”: - o aprendiz deve conquistar o domínio de cada "parte" para que o "segundo todo" seja eficaz. - Se o aprendiz não entender uma das "partes", não poderá haver entendimento completo do todo (p. 237).
- Reconciliação Integrativa (conforme Figura 4)	O segundo “ TODO ”: - considerado o componente mais relevante. - ligar as "partes" novamente, para que formem o todo completo, pois não é apenas o domínio de cada parte do ensino que é importante, mas também a relação entre essas "partes" por meio do "segundo todo" que fornece ao aprendiz o entendimento completo do conteúdo - usar a aprendizagem individual no "segundo todo" permitirá que “os alunos pratiquem todas as suas habilidades em um procedimento contínuo”. (KNOWLES; HOLTON; SWAMSON, 2011, p.235).

Fonte: Da autora (2020).

Em se tratando de adulto, é relevante estimular a responsabilidade social, formando profissionais competentes, com autoestima, seguros de suas habilidades profissionais e comprometidos com suas responsabilidades, críticos e criativos, pois na andragogia não há

²² Estruturas mentais individuais

instrumentos de mensuração da aprendizagem, sendo o ponto forte “demonstrar empiricamente que as técnicas andragógicas levam a melhores resultados” (KNOWLES; HOLTON; SWAMSON, 2011, p. 226). Cabe salientar que as aprendizagens por recepção e descoberta podem ser utilizadas pelo professor em uma mesma experiência de aprendizagem, que deverá escolher a que melhor se ajuste ao momento.

A resolução de problemas é um método prático e válido de analisar a capacidade do aprendiz e, conseqüentemente, avaliar sua aprendizagem significativa, desde que não sejam propostos ou escritos de forma que o sujeito já conheça. Diante disso, estaria o aprendiz fazendo uma simulação de aprendizagem significativa (MOREIRA; MASINI, 2006).

Resumidamente, muitas vezes, o PC é descrito como um método de resolver problemas, portanto, de uma forma significativa, porque uma vez resolvido, estará pronto. Caso novas funcionalidades ocorram (necessidades), será uma nova situação e um novo problema a ser resolvido, que o será baseando-se nas experiências adquiridas e nos subsunçores ancorados.

Consecutivamente, será explicado o que são os cursos binacionais e a sua relação com este trabalho.

2.4 Cursos binacionais

Os sujeitos envolvidos nesta pesquisa foram escolhidos pela particularidade de serem chamados “alunos binacionais”. Essa denominação deriva da matrícula deles em cursos binacionais.

Estes cursos são o resultado de muitas negociações e relações bilaterais entre o governo brasileiro e o uruguaio para as cidades-gêmeas²³ de fronteira. Entre elas, Sant’Ana do Livramento (Brasil) e Rivera (Uruguai). Para a educação, particularmente, foram assinados três acordos importantes que impactaram diretamente na criação destes cursos:

- a) Acordo Básico de Cooperação Econômica, Científica e Técnica entre o Governo da República Federativa do Brasil e o Governo da República Oriental do Uruguai, firmado em 12/06/1975 e promulgado pelo Decreto N° 78.159 de 02/08/1976²⁴;

²³ O Ministério da Integração Nacional define cidades gêmeas como “os municípios cortados pela linha de fronteira, seja essa seca ou fluvial, integrada ou não por obra de infraestrutura, que apresentem grande potencial de integração econômica e cultural, podendo ou não apresentar a unificação da malha urbana com cidade do país vizinho” (BRASIL, Portaria n. 125, 2014, texto digital).

²⁴ Disponível em: <http://legis.senado.leg.br/norma/500799/publicacao/15782514>.

- b) Acordo entre os Governos da República Federativa do Brasil e da República Oriental do Uruguai para Permissão de Residência, Estudo e Trabalho a Nacionais Fronteiriços Brasileiros e Uruguaios, celebrado em Montevideu, em 21 de agosto de 2002²⁵;
- c) Acordo de Criação de Escolas e/ou Institutos Binacionais Fronteiriços Profissionais e/ou Técnicos e para o Credenciamento de Cursos Técnicos Binacionais Fronteiriços²⁶ em abril de 2005, do lado brasileiro, e Ley Nº 18.158 - *Acuerdo para la Creación de Escuelas y/o Institutos Binacionales Fronterizos Profesionales y/o Técnicos y para la Habilitación de Cursos Técnicos Binacionales Fronterizos*²⁷, por parte do governo do Uruguai.

Em 2011, iniciam os primeiros cursos técnicos binacionais, em parceria com a *Consejo de Educación Técnico Profesional - Universidad del Trabajo del Uruguay* (CETP-UTU), de Rivera. Para brasileiros e uruguaios, Controle Ambiental, sob responsabilidade do CETP-UTU, na cidade de Rivera, e Informática para Internet, em Santana do Livramento, sob a direção do Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense (IFSUL), Câmpus Santana do Livramento (CSL). Atualmente existem os cursos: Técnico em Eletroeletrônica, Técnico em Informática para Internet e Técnico em Sistemas de Energia Renovável, na modalidade Integrado e Sistemas de Energia Renovável na modalidade Subsequente (pós-médio). Estes cursos acontecem no Câmpus Santana do Livramento e os técnicos em Logística, Gastronomia/Cozinha e Controle Ambiental na UTU em Rivera, estes últimos todos na modalidade pós-médio.

Um anexo ao convênio original foi firmado entre a *Universidad Tecnológica del Uruguay* (UTEC), CETP-UTU e IFSUL, em 26-11-2018, como objetivo de unir esforços para a execução de atividades de ensino, pesquisa e extensão ou cooperação técnica/tecnológica em geral, priorizando o planejamento e promoção de cursos técnicos, graduação e pós-graduação binacionais relacionadas às demandas de formação técnica e tecnológica da região. Neste documento, foram incluídos os cursos superiores de Análise e Desenvolvimento de Sistemas e Tecnólogo em Mecatrônica Industrial.

No Câmpus Jaguarão (RS, Brasil) é ofertado o curso técnico em Edificações e no lado uruguaio, na cidade de Rio Branco, o curso de Arroz y Pasturas.

²⁵ Disponível em: <https://www2.camara.leg.br/atividade-legislativa/comissoes/comissoes-mistas/cpcms/historico/acordos-mercosul/Anexo%20Dec.%20Leg.%20933%202009%20Acordo%20Brasil%20Uruguai%20Permissao%20de%20Residencia-%20Estudo%20e%20Trabalho%20e%20Nacionais.pdf>.

²⁶ Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2015/decreto/D8455.htm

²⁷ Disponível em: <https://legislativo.parlamento.gub.uy/temporales/leytemp4708003.htm>. Acesso em: 03 jun. 2019.

Como características diferenciais destes cursos, com base em Citolin (2013) e em Silva e Lima (2015), apontam-se:

- o perfil do egresso deve atender a ambas as realidades da fronteira bem como as demandas do setor laboral;
- a construção conjunta do projeto pedagógico, cargas horárias, entre outros;
- a seleção dos candidatos segue a normativa do seu país de nascimento, ou seja, os brasileiros são submetidos a um processo seletivo, e no Uruguai acontece um sorteio de vagas a partir de uma lista de interesse;
- as vagas para os cursos são divididas, igualmente, entre as duas instituições, independente do curso ou de onde este é ministrado, inclusive com reciprocidade das vagas entre as instituições,
- a definição do português e do espanhol como línguas maternas; entre outras.

A fim de comparar as equivalências de organização do ensino, a Figura 9 mostra:

Figura 9 - Equivalência da Organização dos Sistemas Educacionais Brasil e Uruguai

		EDUCAÇÃO BÁSICA			GRADUAÇÃO		POS-GRADUAÇÃO		
BRASIL	EDUCAÇÃO INFANTIL	EDUCAÇÃO DE JOVENS E ADULTOS mínimo: 15 anos de idade	EDUCAÇÃO DE JOVENS E ADULTOS mínimo: 18 anos de idade	EDUCAÇÃO PROFISSIONAL	EDUCAÇÃO PROFISSIONAL	CURSOS DE EXTENSÃO	LATU SENSU	POS-DOUTORADO	
						CURSOS SEQUENCIAIS	ESPECIALIZAÇÃO APERFEIÇOAMENTO OUTROS		
	CRECHE PRÉ-ESCOLA	ANOS DE ESTUDO 9 anos	ANOS DE ESTUDO 3 a 4 anos			ANOS DE ESTUDO 3 A 6	MESTRADO	DOUTORADO	
		ENSINO FUNDAMENTAL	ENSINO MÉDIO		EDUCAÇÃO SUPERIOR Duração variável				
							STRICTO SENSU		
URUGUAY	GUARDERIA	ENSEÑANZA PRIMARIA	CICLO BÁSICO	ESPECIALIZACIONES	FORMACIÓN DOCENTE				
	JARDIN DE INFANTES		CICLO BÁSICO TECNOLÓGICO 3 años de duración	EDUCACIÓN MEDIA TECNOLÓGICA (Escuela Técnico Tecnológica) 3 años de duración					
	JARDINERA		ARTICULACIÓN EDUCACIÓN MEDIA BÁSICA Programa flexible 3 módulos de duración	EDUCACIÓN PROFISSIONAL 1 año de duración	BACHILLERATO PROFESIONAL 3 años de duración	UNIVERSIDAD			
			FORMACIÓN PROFESIONAL BÁSICA Plan 2007 (2,3 años de duración)	ACREDITACIÓN DE SABERES	ING. TECNOLÓGICO CURSOS TÉCNICOS TECNÓLOGOS - TECNATURAS 1 a 4 años de duración				
			CURSOS BÁSICOS ESPECIALIZACIONES Formación Profesional	CAPACITACIÓN	BACHILLERATO DIVERSIFICADO 3 años de duración				
		EDUCACIÓN PRIMARIA	NIVEL 1	NIVEL 2	NIVEL 3				

Fonte: Cordenonzi *et al.* (2014, p. 6)²⁸.

²⁸ CORDENONZI, Walkiria H. *et al.* **Mapeamento da Educação Binacional - Relatório Técnico I.** Santana do Livramento, RS, 2014. Disponível em: <http://ecompartindo.santana.ifsul.edu.br/course/view.php?id=91>. Acesso em: 3 mar. 2019. Design da Figura: Juan Marcelo Botasso.

Na Figura 9, são mostradas as equivalências dos sistemas educativos que figuram neste trabalho e suas correspondências.

Neste espaço binacional, são compartilhados muitos aspectos, elucidados como:

[...] existe una integración social, cultural, comercial, de influencias mutuas, de hábitos compartidos, costumbres que se intercambian, en una hermandad, que siempre está presente, aunque a veces pase desapercibida tras algún evento, como puede ser el deportivo. Es una conjunción de elementos socio-culturales-económicos únicos; familias que poseen algún integrante de una o de otra ciudad, o personas conocidas en el medio como “doble chapa” (CORDENONZI *et al.*, 2014, p. 149).

Pode-se perceber, ao analisar a Figura anterior, que há diferenças entre a organização de ensino de ambos países. No Uruguai, a partir dos 12 anos, quando o estudante termina a “primaria”, já encontra um conjunto de opções que pode escolher para sua carreira. Por outro lado, isso não acontece no Brasil, onde a primeira oportunidade somente se dá no Ensino Médio (por exemplo, ensino profissionalizante), com idade de 15 anos.

A explicação sobre as diferenças existentes entre ambos os países, tanto na questão da estrutura de ensino, quanto na mistura de idiomas, é importante no sentido da forma de comunicação verbal. Segundo Ausubel, Novak e Hanesian (1983), a linguagem é uma das principais formas de comunicação entre professor e aprendiz, tendo um papel relevante de facilitador do processo. Esses autores salientam três pontos fundamentais:

- a) a linguagem reflete o nível de desempenho cognitivo e intervém na aquisição de conceitos;
- b) “o processo de assimilação de conceitos por definição e por contexto seria totalmente inconcebível sem a linguagem” (AUSUBEL; NOVAK; HANESIAN, 1983, p. 101-102, tradução nossa);
- c) “a linguagem contribui para assegurar certa uniformidade cultural no conteúdo genérico dos conceitos, a fim de facilitar a comunicação cognitiva entre as pessoas” (AUSUBEL; NOVAK; HANESIAN, 1983, p.102, tradução nossa).

Em um ambiente educacional – pautado pela preocupação do processo de ensino e aprendizagem – onde se encontram indivíduos com línguas maternas diferentes, é preciso que o professor (independentemente do idioma que está utilizando para se comunicar) se faça entender/compreender pelos alunos, além de ter extensivo cuidado para com os materiais utilizados (apostilas, testes, exercícios, entre outros) com os estudantes.

Uma das habilidades que está sendo proposta nesta tese, que influencia no desenvolvimento do pensamento computacional (PC), é a compreensão (ver definição no Capítulo 4). Neste ponto há dois aspectos a serem considerados: a compreensão do idioma e a

compreensão do problema. Por exemplo, a um aluno uruguaio é apresentado um problema para que desenvolva a solução, escrito na língua portuguesa. Se a solução não estiver completamente correta, a avaliação do erro torna-se complicada, já que o mesmo pode ter sido originado da interpretação do texto (pois não entendeu as palavras ou fez uma tradução equivocada) ou entendeu o texto, porém não desenvolveu a solução corretamente.

Por isso, os testes utilizados nesta tese foram disponibilizados nos dois idiomas “oficiais” dos cursos binacionais, ficando a critério do aluno a escolha da língua na qual será avaliado. A comunicação com os alunos, por parte da autora, se dá em português e em espanhol (com menor frequência). Um outro fator a ser avaliado, ainda relacionado às linguagens, é a linguagem de programação utilizada para desenvolver a habilidade resolução algorítmica (definida no próximo capítulo) e que apresenta sintaxe e semântica própria. Portanto, no decorrer do desenvolvimento deste trabalho, se considera o uso de três linguagens.

A seção seguinte apresenta os trabalhos considerados afins com os assuntos relacionados a esta pesquisa.

2.5 Trabalhos correlatos

Na seção anterior foram apresentados alguns trabalhos sobre PC, APC, teorias de aprendizagem, todos com alguma relação com os temas abordados nesta tese.

Foi realizada uma ampla pesquisa bibliográfica no Catálogo de Teses e Dissertações (CTD)²⁹, segundo a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), em repositórios digitais, tais como *Association for Computing Machinery Digital Library (ACM)*, *Institute of Electrical and Electronics Engineers Computer Science Digital (IEEE)*, *Science Direct*, SCOPUS, anais de eventos, livros, revistas, entre outros; incluindo as áreas de computação, educação e ensino.

As literaturas selecionadas a seguir foram escolhidas e organizadas de acordo com determinados critérios. Primeiramente, são listadas no início desta seção aquelas que apresentam pesquisas sobre PC, com adultos (nível pós-médio), ou utilizaram a ferramenta *App Inventor* e/ou, ainda, o teste *Bebras*.

Um grupo de professores da Unisinos³⁰ relatam que ofertaram uma capacitação sobre PC dirigida somente para colegas que trabalham em cursos superiores, de diversas áreas de atuação.

²⁹ Disponível em: <https://catalogodeteses.capes.gov.br/catalogo-teses/#/>. Acesso em: 20 dez. 2018.

³⁰ Unisinos - Universidade do Vale do Rio dos Sinos, com sede em São Leopoldo, RS.

O objetivo foi de “sensibilizar os professores sobre as possibilidades de aplicação deste modelo de pensamento em atividades educacionais relacionadas a variados contextos de aprendizagem” (KAMPPF *et al.*, 2016, p. 2). Desenvolveram um *template* – Canvas³¹ Algorítmico – para orientar os participantes na solução de problemas, que, no final, reconheceram a importância de trazer o PC para a Universidade e para a formação dos docentes.

No trabalho de Souza, Rodrigues e Andrade (2016), é descrita uma oficina realizada com docentes e discentes em algumas escolas do Serviço Social da Indústria (SESI) no Estado da Paraíba. Seu objetivo foi observar se o processo de ensino-aprendizagem de robótica melhorava com a adição do PC. Dos docentes que participaram da formação, 75% concordam que, embora o conceito de PC ainda não esteja totalmente entendido, o impacto deste foi muito positivo no desempenho dos estudantes. A respeito da idade dos docentes, 50% possuem entre 35 e 45 anos e há citações das ideias de Piaget e Papert no trabalho. Através destas duas citações, pode-se observar que utilizaram sujeitos adultos. Os próximos trabalhos comentados utilizaram o *App Inventor*.

Roscoe, Fearn e Posey (2014) fizeram uma interação com estudantes ingleses para ensinar o PC. Primeiramente, utilizaram o *PrintCraft*³², na sequência o *App Inventor* e no terceiro momento o *Scratch* e *Arduino*³³. Talvez esta seja uma das poucas pesquisas com alunos de Educação Fundamental (EF) utilizando o *App Inventor*. É relatado no trabalho que os alunos não apresentaram problemas na aprendizagem de variáveis, de comandos de condição e laço e de componentes básicos da ferramenta, como, por exemplo: botões e sensores. Também não foi comentada a existência de dificuldade de utilizar o ambiente integrado de desenvolvimento e uso do aplicativo.

Os pesquisadores Gomes e Melo (2013) propuseram uma metodologia embasada na aprendizagem significativa e, como ferramenta de desenvolvimento visual, empregaram o *App Inventor*. Seu objetivo foi apresentar os conceitos de programação de forma mais flexível e atraente, no formato *blended learning*, para um grupo de 41 estudantes do primeiro e segundo ano do Ensino Médio (EM). No final do curso, foi aplicado um questionário para avaliar a eficácia da metodologia, concluindo que as maiores dificuldades dos alunos se deu “sobre três pilares: 1. Leitura e interpretação de texto; 2. Matemática básica; 3. Idioma da ferramenta”

³¹ O canvas é um resumo dos pontos-chave de um planejamento.

³² *PrintCraft* é um servidor web para o jogo *MineCraft* em que o jogador poderá criar seu modelo obtendo depois o arquivo compatível com a impressão tridimensional. Disponível em: <http://www.printcraft.org/>. Acesso em: 22 maio 2018.

³³ O *Arduino* é uma plataforma eletrônica de código aberto baseada em *hardware* e *software* fáceis de usar (tradução nossa). Disponível em: <https://www.arduino.cc/>

(GOMES; MELO, 2013, p. 8), sendo que a totalidade dos alunos concordaram que a metodologia e a ferramenta contribuíram muito para a aprendizagem de forma significativa. Neste trabalho, pôde-se verificar a dificuldade de entender (habilidade compreensão) a descrição do problema. Quanto ao idioma da ferramenta, atualmente já existe a versão tanto em português quanto em espanhol.

Greff *et al.* (2018) utilizaram *App Inventor* com um grupo de 15 adultos (sendo que 80% tinha entre 35 e 54 anos) em somente um encontro de uma hora, no qual desenvolveram um App simples. No encontro utilizaram, de forma experimental, o curso disponibilizado pelo próprio site³⁴ do MIT (*Massachusetts Institute of Technology*). Dentre as conclusões apresentadas, ressalta-se que “se faz necessário exaltar as possibilidades criativas de construção presentes neste tipo de ferramenta e que, por oferecer como diferencial a facilidade de operação, permite um maior número de usuários e conseqüentemente maior diversidade de soluções criativas” (GREFF *et al.*, 2018, p. 320).

Em síntese, os trabalhos citados (GOMES; MELO, 2013; ROSCOE; FEARN; POSEY, 2014) utilizaram o *App Inventor*, porém a faixa etária (e conseqüentemente o objetivo do curso) é diferente da proposta desta tese. Já no trabalho de Greff *et al.* (2018), os autores não desenvolveram nenhum constructo pedagógico para aplicação, bem como o tempo de utilização não pode ser comparado com a proposta desta tese, embora trabalharam com adultos no *App Inventor*.

A metodologia *The Rubric* (SHERMAN; MARTIN, 2015) utiliza conceitos de programação (e PC) com aspectos de mobilidade, dividindo estes conceitos em duas dimensões. A partir da programação de aplicativos no *App Inventor*, os autores analisam a presença ou ausência de conceitos, com base em uma lista de itens. Para cada item, definiram uma escala de 1 a 4. A metodologia foi aplicada em dois grupos de estudantes universitários: em um Curso de Computação e outro curso (não foi citado). Porém, os autores somente analisam a aprendizagem, utilizando a metodologia, a partir do produto final, ou seja, avaliam a presença ou ausência do uso dos conceitos no resultado, não se preocupando com o processo de construção do conhecimento.

No artigo de Laisa, Henrique e Oliveira (2018), foi descrito um estudo exploratório realizado com 19 alunos graduandos distribuídos nos cursos de Tecnologia da Informação, Química, Licenciatura em Filosofia, História e Licenciatura em Física de uma universidade federal. O objetivo foi investigar se as questões do *Bebras* eram apropriadas para que os alunos

³⁴ Disponível em: <http://appinventor.mit.edu/explore/teaching-app-creation.html>. Acesso em: 5 maio 2018.

pudessem construir documentos de *game design*. Apontaram que os alunos produziram os documentos de maneira mais concisa e eficaz devido ao uso do PC.

Na dissertação de Mestrado de Couto (2019), a autora descreveu sua experiência em propor um curso de PC para alunos do EF. Para realizar a avaliação utilizou o instrumento de diagnóstico desenvolvido por Gonçalves (2015). A partir dos resultados obtidos, propôs melhorias no instrumento. Para complementar e validar os resultados encontrados, aplicou o *CT-Test*. Relembrando que este teste foi desenvolvido por Román González, Pérez González e Jiménez Fernández (2015).

No Quadro 3 é mostrada a relação dos temas correlatos com os trabalhos citados.

Quadro 3- Relação dos trabalhos correlatos

Autor(es)	App Inventor	Andragogia	Teste Bebras
(ROSCOE; FEARN; POSEY, 2014)	X		
(GOMES; MELO, 2013)	X		
(SHERMAN; MARTIN, 2015)	X	X	
(GREFF <i>et al.</i> , 2018)	X	X	
(LAISA; HENRIQUE; OLIVEIRA, 2018)		X	X
(SOUZA; RODRIGUES; ANDRADE, 2016)		X	
(KAMPFF <i>et al.</i> , 2016)		X	
Esta pesquisa	X	X	X

Fonte: Da autora (2019).

No Quadro anterior, pode-se perceber o entrelaçamento das pesquisas já publicadas e que apresentam algum ponto convergente com os conceitos que tratam este trabalho: ferramenta *App Inventor*, aprendizagem de adultos e o uso do Teste *Bebras*. Porém nenhum deles aborda, de forma concomitante, os temas desta tese.

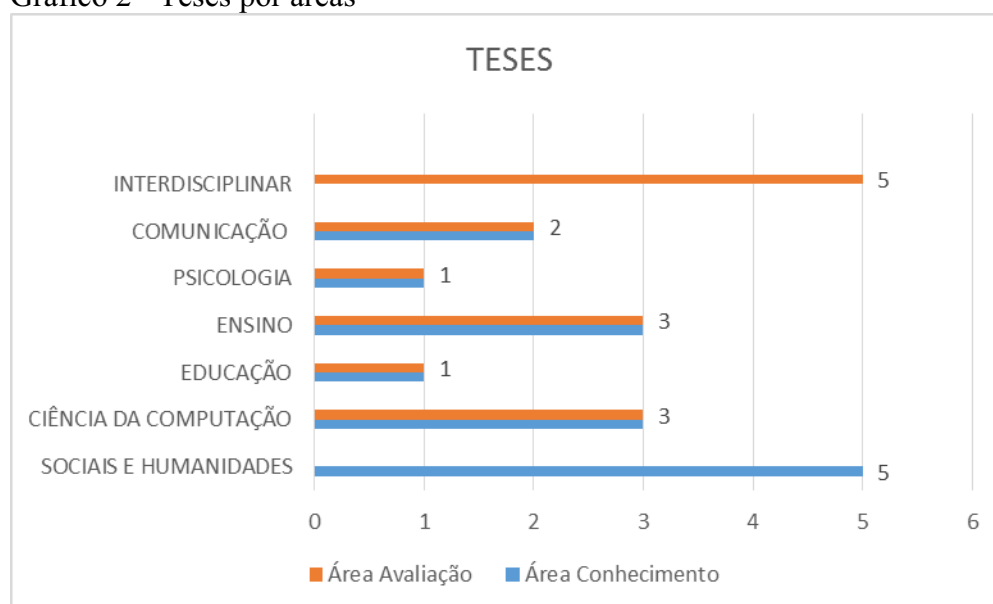
Ao se realizar uma busca no Banco de Teses e Dissertações da CAPES (BTD), com as palavras-chave “pensamento computacional” (em 14-01-2019) resultaram 57 trabalhos, sendo 14 teses de Doutorado e as demais, dissertações de Mestrados, a partir de 2009. Uma segunda pesquisa com as mesmas palavras-chave, no dia 25-04-2020, porém selecionando o ano de 2019 (para fins de atualização), retornaram 14 trabalhos, entre eles, uma tese. Nesta mesma data, uma consulta foi executada para somente “Mestrado” e “Mestrado Profissional” sem especificar um ano determinado. Essa consulta resultou em 72 dissertações. A compilação de ambas as consultas está apresentada a seguir.

2.5.1 Teses

A partir da pesquisa no Banco de Teses e Dissertações, das 15 teses retornadas foram lidos os resumos e a introdução de todas elas. A busca focou em que nível de ensino os autores desenvolveram suas propostas e quais as teorias de aprendizagem em que se embasaram.

No Gráfico 2, estão os dados da pesquisa, organizados por área de conhecimento e por área de avaliação, segundo a classificação da Capes.

Gráfico 2 - Teses por áreas



Fonte: Da autora (2020).

A partir de busca da palavra-chave “Pensamento Computacional”, as teses predominantemente se encontram na área de Ciências Sociais e Humanas, sendo que todas pertencem à área de avaliação Interdisciplinar, ou seja, 33,33%. Na continuação, tem-se a Ciência da Computação e o Ensino, ambos com 3 teses defendidas, perfazendo 20% cada uma.

A fim de sintetizar e identificar os programas dos cursos nos quais as 15 teses foram desenvolvidas, as teorias de aprendizagem referenciadas e os níveis de ensino onde foram aplicadas, é proposta a Tabela 1.

Tabela 1- Dados de Teses defendidas a partir de 2009

Nome do Programa	Número de Teses defendidas	Teoria da Aprendizagem	Nível de Ensino
Educação	2	A	E.M.
		C	E.B.
Informática na Educação	5	B	E.F.
		A	E.F.

Nome do Programa	Número de Teses defendidas	Teoria da Aprendizagem	Nível de Ensino
		A + B	-- ³⁵
		B	E.F.
		A + B + C	E.S.
Psicologia	1	A + B + E	E.S.
Ensino de Ciências	1	A + ABP ³⁶	E.M
Educação em Ciências Químicas da Vida e Saúde	1	B + D	E.F.
Comunicação Social	1	--	--
Ciência da Computação	2	--	E.S.
Informática	1	A + B	E.M; E.S.; EJA ³⁷
Comunicação e Semiótica *	1	--	--

Fonte: Da autora (2020).

Legenda: A: Construcionismo de Papert; B: Construtivismo de Piaget; C: Aprendizagem Significativa de Ausubel; D: Fluidez Digital de Resnick; E: Vygotsky.

Das teses analisadas, 57,14% (8) utilizaram como teoria de aprendizagem o construcionismo. Somente uma tese foi desenvolvida utilizando exclusivamente a teoria de Piaget. Já a aprendizagem significativa aparece apenas em dois trabalhos, um no nível fundamental e outro no superior (AMARAL, 2015).

São os programas de pós-graduação em Informática na Educação que lideram as teses defendidas. Isso quer apontar que há a união da Ciência da Computação aplicada à Educação. São apenas dois autores que se aventuraram na Educação a tratar sobre este tema. Apenas 5 teses tiveram como sujeitos cursantes no Ensino Superior, que corresponde a 26,66%, e uma na Educação de Jovens e Adultos (EJA). O resumo destas teses está apresentado a seguir.

A tese de Santanna (2014) focou seu estudo na aprendizagem de conceitos da Computação e de PC para alunos do Curso Superior de Design e o seu produto foi uma proposta pedagógica e uma linguagem de programação denominada de *RocketSocket*. A autora fundamentou seu trabalho na Psicologia socio-histórica (Vygotsky), na Epistemologia Genética (Piaget) e em Jean-Claude Abric para as representações sociais. Por outro lado, não realizou avaliação da aprendizagem do PC.

A proposta de uma ferramenta para auxiliar na compreensão de código-fonte é o trabalho de Müller (2017). Os sujeitos da pesquisa são indivíduos iniciantes em programação (Ensino Superior) que usam e reusam seus próprios códigos-fonte ou de outros programadores. A autora faz observações sobre o uso da ferramenta e sobre os aspectos comunicativos passados pelos

³⁵ Esta tese ainda não está disponível na data da pesquisa, no resumo cita adolescentes, que podem estar no E.M. ou E.F.

³⁶ ABP - Aprendizagem Baseada em Problemas.

³⁷ EJA - Educação de Jovens e Adultos.

códigos, embasando-se na Engenharia Semiótica. Não menciona nenhuma teoria de aprendizagem e processo avaliativo de apropriação de PC.

Frota (2017), em sua tese “Arcabouço Conceitual de Adaptação de Recursos Educacionais”, aborda a proposta de um Curso de PC e Algoritmos através do uso de AVA na modalidade semipresencial, utilizando sistemas adaptativos de recursos, baseados no construtivismo. O modelo proposto foi de Arquiteturas Pedagógicas Multiagente Adaptativas para Recursos e Atividades em AVA. Este modelo permite que o aluno acesse seu estado atual do curso, proporcionando uma aprendizagem auto-regulada. O pesquisador concluiu, a partir deste modelo, que o nível de aprendizagem dos alunos, no pré e pós-teste, evoluiu significativamente em todas as habilidades avaliadas: simulação, domínio em programação; domínio em estruturas condicionais; domínio em estruturas de repetição; abstração; domínio em projeto de algoritmos; reconhecimento de padrões e análise de dados e decomposição de problemas.

No seu Doutorado em Ensino de Ciências, Barcelos (2014) trabalhou com alunos do EM do Brasil e do Chile, mapeando as habilidades e competências relativas à computação, bem como um mapeamento destas entre PC e Matemática. Utilizou oficinas de jogos e concluiu que a motivação dos alunos foi um importante ponto a ressaltar e propõe diretrizes para a criação de atividades com o mesmo fim. Esta tese está classificada no item “Ensino de Ciências”.

Em sua tese “Processo de ensino e aprendizagem de algoritmos integrando ambientes imersivos e o paradigma de blocos de programação visual”, Amaral (2015) propõe uma prática pedagógica na qual associa ambientes imersivos a paradigmas de programação, para estudantes de cursos superiores, motivado, entre outros, pelos expressivos índices de reprovação e evasão das disciplinas de Algoritmos (que envolve PC e resolução de problemas). Essa prática resultou promissora no que se refere ao desenvolvimento do raciocínio lógico e na “melhoria do seu desempenho acadêmico”, sendo fundamentada nas teorias de Piaget, Papert e Ausubel, na área de Informática na Educação.

Soster (2018) focou sua pesquisa na Educação Maker, na área de Língua Portuguesa, como forma de desenvolver o processo de ensino e aprendizagem em TDIC e de PC, no EB. Como resultado, apresenta um conjunto de práticas utilizando a Educação Maker, sendo que a área de desenvolvimento de sua tese foi a Educação.

Na sua tese, Araújo (2019) se propôs a investigar estratégias e instrumentos para quantificar o PC sem o uso de práticas de programação. Como resultado, desenvolveu um modelo baseado em estudos empíricos para quantificar o PC como uma habilidade cognitiva. Este modelo é dividido em quatro competências e doze habilidades. Fez uso do *Bebras* e os sujeitos

envolvidos foram alunos da Universidade Federal de Campina Grande, do Curso de Bacharelado em Ciência da Computação, e da Universidade Federal de Pernambuco, matriculados nos Cursos de Bacharelado em Sistemas de Informação e Licenciatura em Ciência da Computação.

A partir das 15 teses retornadas, 6 foram lidas completamente, porque havia semelhança de sujeitos atuantes no ES e/ou Área de Avaliação e/ou Teoria proposta por Ausubel com o interesse deste trabalho. Estas estão elencadas no Quadro 4.

Quadro 4 - Resumo das Teses X Área

Autor	Nível de Ensino	Ferramenta utilizada	Foco/ disciplina	Tipo de pesquisa	Área
(SOSTER, 2018)	EB	---	Educação Maker	Qualitativa	Educação
(AMARAL, 2015)	ES	Ambientes imersivos	Algoritmos	Quali-quantitativa	Informática na Educação
(BARCELOS, 2014)	EM	Construção de Jogos	PC e Matemática	Etnografia fenomenológica – (Qualitativa)	Ensino de Ciências
(MÜLLER, 2017)	ES	Código-fonte	Programação	Qualitativa	Ciência da Computação
(FROTA, 2017)	ES	AVA	Aprendizagem auto-regulada	Qualitativa	Informática
(SANTANNA, 2014)	ES	Linguagem Rocket-Socket	Princípios da Computação	Triangulação (Quali)	Psicologia
(ARAÚJO, 2019)	ES	Teste Bebras	Avaliação do PC	Quali-quantitativa	Ciência da Computação

Fonte: Da autora (2020).

Para o constructo da proposta desta tese, embora o construcionismo não tenha sido o foco – pois é possível perceber que a teoria evidencia o desenvolvimento cognitivo das crianças – nota-se as semelhanças e a importância de desenvolver o pensamento computacional, tanto em crianças como em adultos. Em nenhum trabalho foram encontradas referências ao uso do *App Inventor*.

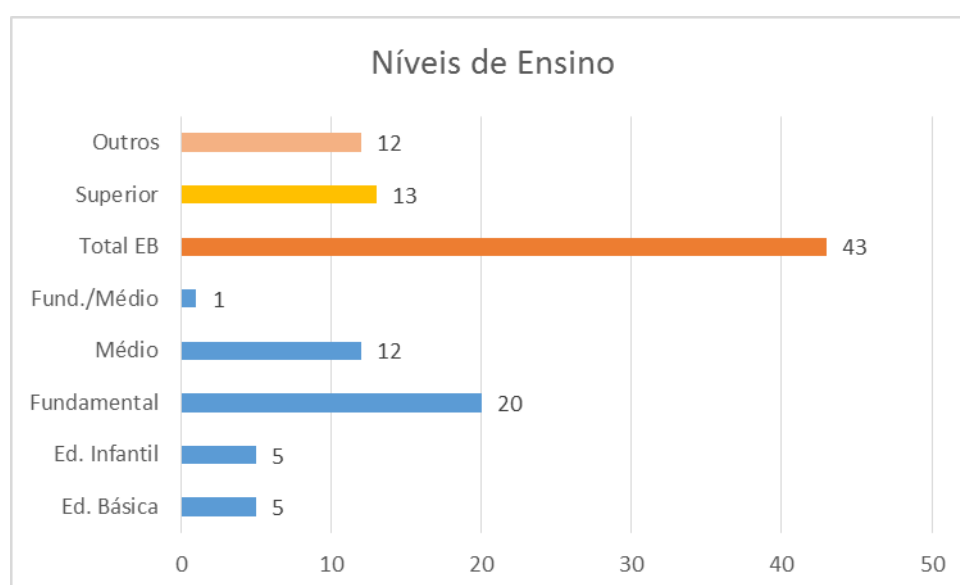
Pode-se constatar que a tese de Amaral (2015) é a que mais se aproxima da proposta deste trabalho: o autor utiliza a teoria de aprendizagem significativa (entre outras), no ensino superior (adultos), porém sua abordagem focou na programação e ambientes imersivos. O conjunto das teses e suas classificações estão listadas no Apêndice A.

2.5.2 Dissertações

Seguindo a pesquisa no Catálogo de Teses e Dissertações da Capes, figuraram 72 Dissertações que continham as palavras “pensamento computacional”. Do total foram desconsiderados três trabalhos, porque as palavras de busca apareceram somente como dados do contexto, isto é, na Linha de Pesquisa.

Dos restantes, em 80% (55) dos casos aparece o termo de busca no item Palavras-Chave; são 13 trabalhos os quais tem seu foco no Ensino Superior³⁸ e 8 estão na Área de Ensino. As divisões das Dissertações com relação aos níveis de ensino estão elencadas no Gráfico 3.

Gráfico 3 - Número de Dissertações X Níveis de Ensino



Fonte: Da autora (2020).

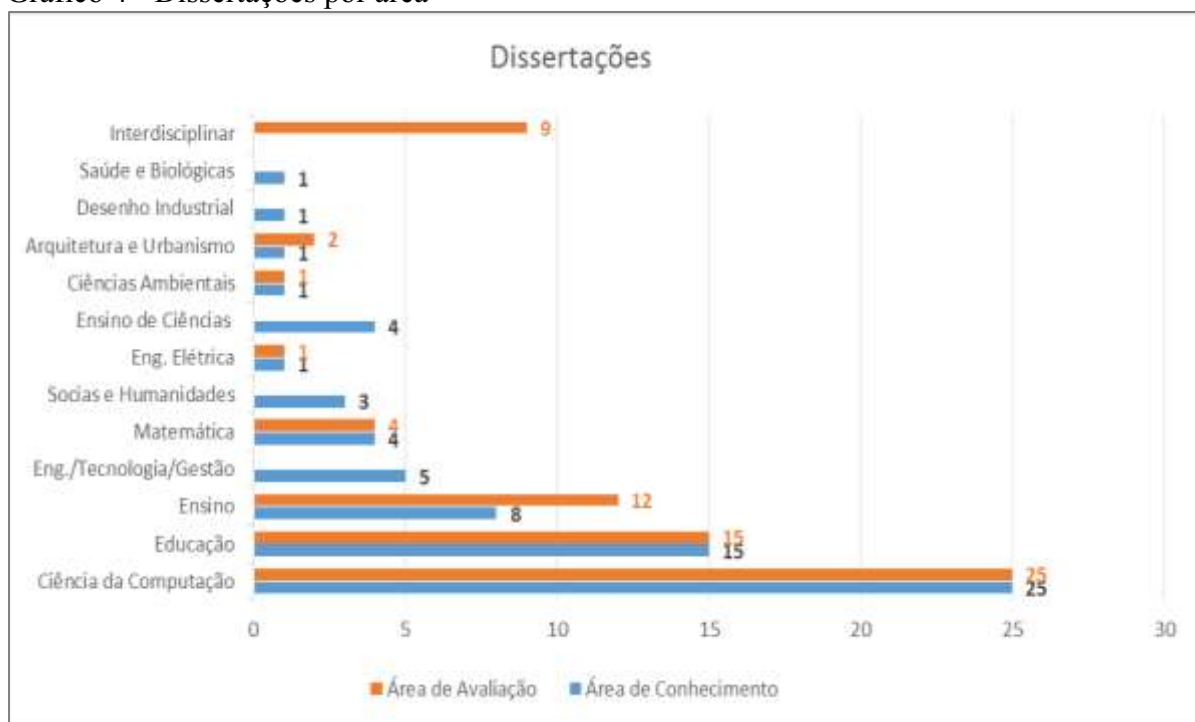
A coluna “Fund./Médio” (Gráfico 3) indica que o trabalho foi realizado nos dois níveis, ou seja, no Fundamental e no Médio. A coluna “Total EB” apresenta o somatório de todas as pesquisas realizadas na Educação Básica, as quais estão representadas na cor azul no Gráfico 3. Na coluna “Outros” estão contabilizadas as Dissertações que propõem mudanças nos currículos, propostas de metodologias, foi aplicada no Ensino Profissionalizante, entre outros.

Pode-se perceber que o interesse da maioria dos pesquisadores está no Ensino Fundamental, seguido pelo Ensino Médio. Essa tendência com relação à faixa etária, no intervalo entre 6 e 18 anos, acompanha as publicações internacionais. Corroborando com os dados, obteve-se que 62,3% das Dissertações se encontra neste nível de ensino.

³⁸ O EB está dividido em Educação Infantil, EF e EM. O primeiro abarca crianças com idade inferior a 6 anos. O EF abrange a idade entre 6 a 14 anos; o EM entre 15 e 17 anos; o ES e a EJA a partir dos 18 anos (SOSTER, 2018).

As Dissertações foram classificadas quanto a sua área de conhecimento e área de avaliação (Gráfico 4). O detalhamento destes dados encontra-se no Apêndice B.

Gráfico 4 - Dissertações por área



Fonte: Da autora (2020).

Conforme pode-se perceber no Gráfico 4, a área de Ciência da Computação é a que mais exibe trabalhos sobre PC, diferentemente das teses, que fica em segundo lugar. A área de Educação aparece em segundo lugar com 21,7%. A avaliação Interdisciplinar, tanto das Dissertações (Gráfico 4) quanto nas Teses (Gráfico 2), sugere indícios da aplicação do PC em diversas áreas, como, por exemplo, em Arquitetura e Saúde.

Os trabalhos desenvolvidos no Ensino Superior estão descritos a seguir.

A pesquisa de Gonçalves (2015) apresenta um modelo de diagnóstico, no formato de jogo *puzzle*, para avaliação do PC, aplicado em todos os níveis de ensino, e no resultado estatístico foram “encontrados indícios onde estudantes que tiveram maior contato com elementos da computação tiveram melhor desempenho diagnosticado pelo instrumento desenvolvido” (GONÇALVES, 2015, p.7). Este instrumento é apresentado como um teste onde o respondente utiliza elementos do PC para resolver problemas de lógica.

Em seu trabalho, Bozolan (2016) apresenta os resultados de avaliação de duas disciplinas semelhantes, sendo uma delas a de PC para uma turma de Curso Superior. O foco das disciplinas foi o entrelaçamento entre PC e Matemática. Concluiu que as ferramentas digitais, incluindo as

linguagens de programação, “oferecem vantagens significativas no processo de aprendizagem, mediante acompanhamento docente” (BOZOLAN, 2016, p. 112). Essa autora utilizou a ferramenta *Processing*³⁹.

Com a proposta de capacitações *online* e o uso do *Scratch* e *Code.org*, o trabalho de Silva Júnior (2018) utiliza a teoria de Vygotsky, com a finalidade de investigar indícios do desenvolvimento de habilidades lógico-matemáticas. Encontrou como resultado a melhoria do raciocínio lógico-matemático e a resolução de problemas, concluindo que o uso dessas ferramentas auxiliou neste processo. Participaram dessa pesquisa sujeitos com idade superior a 18 anos.

O trabalho de Pasqual Júnior (2018) está calcado na formação de professores, utilizando a teoria de Piaget, Papert e pedagogia relacional⁴⁰, na plataforma *Code.org*. Teve como objetivo avaliar as concepções de ensino e aprendizagem resultantes do uso da plataforma e propôs indicadores para uma nova proposta de formação docente. Concluiu que “o pensamento computacional tem se constituído não apenas como uma competência para os profissionais das áreas técnicas, mas, sobretudo, como uma nova competência para o professor do século XXI” (PASQUAL JÚNIOR, 2018, p.114).

Fernandes (2018) também trabalhou com ensino de PC, por meio da Computação Desplugada e o *Scratch*, e o desenvolvimento de habilidades na formação continuada de professores que atuam no Ensino Fundamental. Finalizou afirmando que, embora faltem pesquisas para unificar o PC com os currículos presentes, os docentes “almejam e percebem a aplicação prática do Pensamento Computacional em situações de aprendizagem em sala de aula” (FERNANDES, 2018, p. 5).

Em seu trabalho, Santana (2018) avalia a motivação e a aprendizagem de estudantes de Engenharia Civil, utilizando o *Scratch* e a linguagem de programação *Python*. O autor considerou o modelo Atenção, Relevância, Confiança e Satisfação (ARCS) e como o PC pode ser ensinado em disciplinas introdutórias.

Por sua vez, Santos (2018) focou seu estudo no ensino e aprendizagem de algoritmos no Ensino Superior para estudantes ingressantes na Computação. Utilizando-se da construção de jogos, concluiu que há “correlação positiva entre a capacidade de aprender por meio de conteúdo

³⁹ “O *Processing* é um ambiente e uma linguagem de programação *open-source* para pessoas que querem criar imagens, animações e interações” (BOZOLAN, 2016, p.12).

⁴⁰ Nesta pedagogia, “valoriza-se a interação entre os sujeitos, as relações de troca e a ideia de que o professor não é o centro do conhecimento, mas é, por outro lado, um sujeito que provocará a construção do conhecimento” (PASQUAL JÚNIOR, 2018, p. 74).

gamificado e a melhoria no desempenho dos alunos em conteúdo de algoritmo e lógica aplicada à computação” (SANTOS, 2018, p. 5).

Já Silva (2016), em sua pesquisa, utilizou os conceitos de jogos para auxiliar no ensino de algoritmos e promover o engajamento do estudante. Assim, propôs um modelo de avaliação formativa, com *feedback* da aprendizagem, e concluiu que, a partir do uso deste modelo, aumentou o índice de engajamento dos indivíduos. Realizou seus experimentos em quatro turmas de programação de um Curso de Sistemas de Informação e uma turma no Curso de Licenciatura em Programação.

Na sua dissertação de Mestrado, Crovador (2019) descreve um experimento com programação em um Curso de Engenharia Civil. O objetivo do trabalho foi, entre outros, estimular o PC através da programação. A avaliação dos resultados foi realizada a partir de um questionário e resultou na aprovação da experiência por parte dos alunos, além de perceberem a necessidade de estimular o PC e a programação.

A pesquisa sobre alguns aspectos do PC em graduandos de Matemática foi o tema da dissertação de Barbosa (2019). O desafio foi o desenvolvimento de fractais, através de uso de um *software*, observando quais as habilidades do PC apresentaram. Concluiu que há indícios de autocomplementação das mesmas, ou seja, identificou que “as habilidades completam uma a outra e que tais aspectos podem se manifestar em conjunto” (BARBOSA, 2019, p. 8).

Das referências citadas, duas utilizaram a teoria de Vigostky (BOZOLAN, 2016; SILVA JÚNIOR, 2018) e Pasqual Júnior (2018) baseou seu trabalho em Piaget e Papert. No que tange às demais não foram encontrados dados a esse respeito, inclusa a dissertação na área de Ensino, de autoria de Fernandes (2018), ou de Crovador (2019), na Educação.

Na pesquisa de dissertações foram incluídos os trabalhos classificados na área de avaliação, definido pela CAPES, de Ensino, sendo acrescidos mais três (CARVALHO, 2018; LUMMERTZ, 2016; SILVA, 2018), além de Fernandes (2018). Na Quadro 5 estão indicados alguns dados relevantes sobre as pesquisas de PC desta base de dados.

Quadro 5 - Dados das Dissertações em Ensino

Autor	Nível de Ensino	Ferramenta utilizada	Foco/ disciplina	Tipo de pesquisa	Teoria de aprendizagem
(FERNANDES, 2018)	Formação de professores	Scratch e PC Desplugado	Matemática	Quali	Não informado
(LUMMERTZ, 2016)	EF	Scratch	PC + Matemática + Literacia Digital	Quali	Construcionismo

Autor	Nível de Ensino	Ferramenta utilizada	Foco/ disciplina	Tipo de pesquisa	Teoria de aprendizagem
(CARVALHO, 2018)	EM	Scratch	PC + matemática	Quali	Construcionismo
(SILVA, 2018)	EF	Robótica com Arduino + Scrach for Arduino	Matemática	Quali	Vygostky

Fonte: Da autora (2019).

A partir da observação do quadro anterior, pode-se perceber que todos os trabalhos analisados sob o escopo da área de avaliação “Ensino” são do tipo Qualitativos, abordando o PC juntamente com o componente curricular Matemática. E na sua totalidade utilizaram a ferramenta *Scratch* independente da teoria de aprendizagem que embasou o trabalho, considerando que o *Scratch* sustenta-se na teoria de Papert (construcionismo), com a diferença que a interface do primeiro ficou mais amigável ao usuário e oferece alguns recursos midiáticos (LUMMERTZ, 2016).

Em resumo, a teoria de aprendizagem de adultos, proposta por Knowles (1981), utilizada nesta tese, não foi encontrada em nenhuma referência. Já a teoria da Aprendizagem Significativa de Ausubel é referenciada em duas teses (uma em EB e outra no ES) e em nenhuma dissertação.

Retomando o Quadro 4, percebeu-se que somente 21,43% (3) das pesquisas fizeram seus experimentos no nível superior. É importante ressaltar que nos trabalhos selecionados não houve referências ao ensino pós-médio. Conclui-se que as produções sobre o PC, a partir do nível pós-médio, são limitadas. Portanto, a contribuição desta tese pode ser considerada relevante sob este ponto de vista.

Na continuação, o próximo capítulo aborda temas relacionados aos vários tipos de alfabetizações.

3 ALFABETIZAÇÃO: CIENTÍFICA, DIGITAL, EM CÓDIGO

O objetivo deste capítulo é apresentar uma discussão sobre alfabetização e letramento, partindo de conceitos sobre Alfabetização Científica (AC), aprofundando na Alfabetização Científica e Tecnológica (ACT), Alfabetização Digital (AD), nas quais se discute o uso das Tecnologias Digitais de Informação e Comunicação (TDIC), e mais especificamente na área da Ciência da Computação (CC) sobre Alfabetização de Código (ACod). Como resultado do trabalho desenvolvido e exposto neste capítulo, obteve-se a proposta de novos olhares para os conceitos de Alfabetização de Código e de Letramento em Código, sob o ponto de vista da CC. Essas definições, contribuição inédita da tese (CORDENONZI *et al.*, 2020), serviram como ponto de partida para determinar o constructo avaliativo para desenvolvimento do PC, apresentado no Capítulo 4.

Assim, este capítulo está organizado em três partes: na seção 3.1 discute-se a Alfabetização Científica e Tecnológica (ACT) sob diferentes aspectos e autores; na seguinte, o foco se volta para a Alfabetização Digital; já na outra seção, o escopo está na Alfabetização em código (ACod). Na continuação, reflexões são descritas acerca do tema abordado.

3.1 Alfabetização Científica

A ciência começa a ser ensinada no Brasil como um componente curricular, no século passado, mais especificamente a partir de 1930. Nos EUA e na Europa, já havia sido incorporada no século XIX (SANTOS, 2007). A partir do dicionário Aurélio⁴¹, a palavra

⁴¹ Disponível em: <https://dicionariodoaurelio.com/Ciencia>. Acesso em: 12 set. 2018.

Ciência deriva do latim *scientia* (traduzido por "conhecimento") e refere-se a qualquer conhecimento ou prática sistemáticos, baseado no método científico.

A partir dessa inclusão no currículo escolar, torna-se importante conceituar Alfabetização Científica (AC) e Letramento Científico (LC), do inglês *scientific literacy*, no contexto da didática das Ciências. Porém, encontram-se muitos autores que adotam diferentes definições e entendimento para descrever o tema, como alfabetização científica, alfabetização científica e tecnológica, letramento ou enculturação científica (SASSERON; CARVALHO, 2011). Deparam-se na literatura, além de entendimentos diferentes, traduções diferentes. Nos documentos da UNESCO (sigla em inglês que significa Organização das Nações Unidas para a Educação, a Ciência e a Cultura), por exemplo, o termo *literacy* (de *scientific and technological literacy*) é traduzido pela palavra "cultura", e não pela palavra "alfabetização" (SASSERON; CARVALHO, 2011, p. 3), em cujo entendimento "a alfabetização deve desenvolver em uma pessoa qualquer a capacidade de organizar seu pensamento de maneira lógica, além de auxiliar na construção de uma consciência mais crítica em relação ao mundo que a cerca"(SASSERON; CARVALHO, 2011, p. 3).

Uma das interpretações sobre o tema é sugerida por Miller (1983, p. 3, tradução nossa), o qual defende que "a alfabetização científica se refere à capacidade do indivíduo de ler, compreender e expressar uma opinião sobre assuntos científicos".

Dada a complexidade de conceitos, Kleiman (1995, p. 19) define LC como o "conjunto de práticas sociais que usam a escrita enquanto sistema simbólico e, enquanto tecnologia, em contextos específicos para objetivos específicos".

Já Bocheco (2011), na sua dissertação de Mestrado, comenta que há divergências no campo da linguagem que diferencia o entendimento sobre alfabetização e letramento científico: "o primeiro remete à simples habilidade de ler e escrever, enquanto o segundo refere-se à condição ou estado de quem sabe ler e escrever" (BOCHECO, 2011, p. 72). Ainda, esse autor extrapola a definição de letramento para o âmbito da ciência, apresentando a conclusão de que ser letrado cientificamente significa não só saber ler e escrever sobre ciência, mas também cultivar e exercer práticas sociais envolvidas com a ciência.

Pode-se perceber, a partir das definições de Kleiman (1995) e Bocheco (2011), que ambos salientam sobre a capacidade dos indivíduos de aportarem aos conhecimentos adquiridos em suas *práticas sociais*. Por outro lado, no entendimento de Xavier (2015), o indivíduo *alfabetizado* é aquele que lê e escreve apenas textos simples (por exemplo uma lista de compras), ainda não conseguindo aventurar-se em práticas socioculturais. Apesar de ter cursado a escola, não consegue entender um fato, emitir opinião, ou descrever com clareza uma pessoa

ou situação. Para esse autor: “A capacidade de enxergar além dos limites do código, fazer relações com informações fora do texto falado ou escrito e vinculá-las à sua realidade histórica, social e política são características de um indivíduo plenamente letrado” (XAVIER, 2015, p. 2).

Já para a pesquisadora e linguista Magda Soares, para a ação somente de ensinar a ler e a escrever usa-se a palavra alfabetização, já a palavra “letramento” refere-se ao “estado ou condição de quem não apenas sabe ler e escrever, mas cultiva e exerce práticas sociais que usam a escrita” (SOARES, 1998, p. 47). Pode-se inferir, a partir do entendimento desses conceitos, que uma pessoa alfabetizada pode não ser letrada (lê, mas não consegue interpretar o texto), e vice-versa. A partir dessa definição, os conceitos sobre *letramento* se igualam entre Xavier (2015), Bochecho (2011) e Soares (1998).

Já para Cunha (2018, p. 34), há uma significativa diferença entre ACT e LCT, quando explica que, no conceito de “[...] alfabetização, consideram fundamental o ensino de conceitos científicos, já os que optam por letramento priorizam, no ensino, a função social das ciências e das tecnologias e o desenvolvimento de atitudes e valores em relação a elas”. O autor complementa que o letramento incorpora complexidades diferentes no que tange à escrita e traduz a expressão *Science literacy* por “letramento em ciência” (p. 34).

Considerando as diferenças e semelhanças apontadas pelos autores citados anteriormente, e a existência de muita literatura que pode ser considerada sobre esse tema, esta tese seguirá os conceitos de Alfabetização Científica (AC) e Letramento Científico (LC) propostos por Xavier (2015) e Soares (1998). Esta pesquisa não tem como objetivo explorar minuciosamente esses termos e seus significados, mas sim de retomar (revisar) esses conceitos importantes. Mais detalhes sobre o desenvolvimento histórico podem ser obtidos em Sasseron e Carvalho (2011).

Com relação ao contexto tecnológico, a UNESCO sugere que, considerando o ponto de vista *social* sobre a tecnologia, observa-se que a qualidade de vida da população em geral depende cada vez mais da tecnologia e das tomadas de decisões acerca dela. Cajas (2001) chama a atenção para a inclusão da temática “tecnologia” nos currículos como uma forma de aporte e contribuição à educação científica.

Entendendo a importância da Alfabetização Científica e Tecnológica (ACT) para o indivíduo, torna-se relevante também a compreensão dos diversos recursos tecnológicos que estão à disposição para o uso, de acordo com os interesses e necessidades de cada um. Na próxima seção, a alfabetização digital está sendo explorada.

3.2 Alfabetização Digital (AD)

A sociedade atual, pode-se afirmar, está imersa na tecnologia, praticamente em todos os seus setores. Alguns mais, outros menos, mas, de alguma forma, a tecnologia já atinge a comunidade e modifica a dinâmica do dia a dia. A humanidade está caminhando para a sociedade 5.0, a qual pode ser entendida como sendo motivada e modificada pela tecnologia acelerada, pela mudança social, pela globalização, inovação e a busca do bem comum. Ademais, a tecnologia é distribuída em tempo real, pelo trabalho colaborativo, cocriativo e mobilidade da aprendizagem em qualquer tempo e lugar. Enquanto isso, interagindo nesta sociedade, o sujeito necessita ser alfabetizado digitalmente. Uma pessoa, com apenas um dispositivo móvel (por exemplo um aparelho celular), pode obter diferentes caminhos para chegar a um determinado lugar, estudar, conversar com outras pessoas, pagar suas contas, contar os passos diários, entre outras situações. Os exemplos se limitam à quantidade de *App*⁴² que estão instalados no dispositivo.

Há vários conceitos e entendimentos sobre a Alfabetização Digital (AD), proveniente do inglês *digital literacy*. O objetivo desta seção não é fazer um estudo exaustivo sobre o tema, mas apresentar alguns autores que abordam o assunto.

Para Machado (2012, texto digital), AD é compreendida como a capacidade de utilizar as Tecnologias de Informação e Comunicação (TIC) de maneira que, a “[...] partir das ferramentas encontradas, melhorar o desempenho, a ação e a condição de trabalho e realização” do indivíduo.

No Portal Educação⁴³, a definição é ainda mais restrita sobre AD: “[...] educar e formar sujeitos para a utilização da Internet de forma correta, consciente, ética e com princípios de cidadania” (texto digital). Bonilla (2011, p. 31) disserta sobre o Programa *SocInfo* (Sociedade da Informação)⁴⁴ e sobre o Livro Verde (TAKAHASHI, 2000), este patrocinado pelo governo brasileiro, que apresenta o conceito de AD como sendo “a aquisição de habilidades básicas para o uso de computadores e da Internet” e complementa que o processo de AD capacita “[...] as pessoas para a utilização dessas mídias em favor dos interesses e necessidades individuais e comunitários, com responsabilidade e senso de cidadania” (p. 31). Completa ainda que:

⁴² *App* significa aplicativo ou programa.

⁴³ Disponível em: <https://www.portaleducacao.com.br/conteudo/artigos/conteudo/alfabetizacao/29070>. Acesso em: 14 set. 2018.

⁴⁴ O programa *SocInfo* foi instituído em 1999 pelo governo federal, concebido a partir de um estudo conduzido pelo Conselho Nacional de Ciência e Tecnologia (CCT).

[...] educar em uma sociedade da informação significa muito mais que treinar as pessoas para o uso das tecnologias de informação e comunicação: trata-se de investir na criação de competências suficientemente amplas que lhes permitam ter uma atuação efetiva na produção de bens e serviços, tomar decisões fundamentadas no conhecimento, operar com fluência os novos meios e ferramentas em seu trabalho, bem como aplicar criativamente as novas mídias, seja em usos simples e rotineiros, seja em aplicações mais sofisticadas. Trata-se também de formar os indivíduos para “**aprender a aprender**”, de modo a serem capazes de lidar positivamente com a contínua e acelerada transformação da base tecnológica (BONILLA, 2011, p. 45, grifo nosso).

Independente da formação profissional, Bonilla (2011) aponta a relevância da *fluência* em TIC, entendida como sendo o uso das ferramentas digitais que leva o indivíduo ao exercício da cidadania e salienta o papel importante que exerce a educação no sentido dessa formação.

Para Avello Martínez *et al.* (2013, p. 431, tradução nossa), o conceito de AD está de acordo com o momento histórico: na era digital, integrada aos dispositivos móveis e redes de comunicação, “[...] gerou novas formas de acessar, construir e comunicar o conhecimento [...]”. No entendimento da *Cornell University* (2015, tradução nossa, texto digital), a AD é “a habilidade de encontrar, avaliar, utilizar, compartilhar e criar conteúdo usando as tecnologias da informação e a Internet”.

Por outro lado, a expressão Alfabetização Mediática e Informacional (AMI) é proposta como sendo o centro de aprendizagem para toda a vida e é definida como forma de “[...] empoderar as pessoas em todos os aspectos da vida, para buscar, avaliar, utilizar e criar informação de uma forma eficaz, para alcançar suas metas pessoais, sociais ocupacionais e educativas [...]” (WILSON *et al.*, 2011, p. 16, tradução nossa).

Ainda, há três aspectos da AD que fazem parte dos documentos da UNESCO:

- a) primeiro:** uso da tecnologia (manipulação de ferramentas de escritório e internet);
- b) segundo:** a compreensão crítica (compreensão e avaliação dos conteúdos acessados);
- c) terceiro:** a criação e comunicação/publicação de conteúdos digitais.

Outros autores não usam a sigla AD, de Alfabetização Digital, mas sim LD (Letramento Digital). Já para Xavier (2015, p. 2) ser letrado digital é “[...] assumir mudanças nos modos de ler e escrever os códigos e sinais verbais e não-verbais, como imagens e desenhos [...]”. O autor esclarece que é um “novo jeito de aprender”, no qual o professor já não é mais o ator principal.

A ideia de Valverde Berrocoso (2012) é semelhante à concepção proposta pela UNESCO, no sentido de que as tecnologias devem ser de aprendizagem para a vida. Esse pesquisador argumenta que as TIC não são estáveis (no sentido que se alteram rapidamente) e podem ser utilizadas de diferentes formas e objetivos, sendo, assim, essenciais para informar, apreender e comunicar-se.

A afirmação de que o indivíduo deve dominar o letramento alfabético para que possa atingir o letramento digital, sendo essa uma condição imprescindível, é sustentada por Xavier (2015) e Valverde Berrocoso (2012).

No seu trabalho, Freitas (2011, p. 339) faz um levantamento sobre LD, e no final define como sendo o que segue:

Conjunto de competências necessárias para que um indivíduo entenda e use a informação de maneira crítica e estratégica, em formatos múltiplos, vinda de variadas fontes e apresentada por meio do computador-internet, sendo capaz de atingir seus objetivos, muitas vezes compartilhados social e culturalmente.

Não obstante o pensamento de Freitas (2011), os autores Soares (1998) e Barton e Hamilton (2012) defendem que não existe apenas um tipo de letramento, e sim que há letramentos digitais. Além disso, esses dependem das práticas e mídias utilizadas. Barton e Hamilton (2012) descrevem que o LD é apenas mais um tipo que está presente na sociedade imersa de TDIC.

No Quadro 6 é disposto um resumo dos conceitos apresentados pelos autores citados. Na coluna Nomenclatura está descrita a forma como o autor denomina seu conceito. Na coluna Descrição é apresentado o resumo do conceito, ou seja, o foco que o autor quer ressaltar. Na coluna Conhecimento estão listadas as habilidades e/ou competências que o indivíduo deve exibir. Já na última, está elencado o “entendimento” dos autores sobre o tema.

Quadro 6 - Resumo Conceitual sobre AD e LD

Nomenclatura	Autor	Descrição	Conhecimento	Educação
AD	Machado (2012)	Ferramentas		
AD	Portal Educação	Internet		
AD	SocInfo (BONILLA, 2011)	Internet, computador		Aprender para aprender
AD	Avello Martínez <i>et al.</i> (2013)		Acessar, construir e comunicar	
AMI	UNESCO (WILSON <i>et al.</i> , 2011)	Ferramentas	Buscar, avaliar, utilizar e criar	Educação para a vida Práticas sociais
AD	Cornell University (2015)	Internet	Buscar, avaliar, utilizar, compartilhar	
LD	Xavier (2015)		Escrita	Prática social
LD	Valverde Berrocoso (2012)		Habilidade técnicas	Educação para a vida
LD	Soares (1998)		Escrita	Práticas sociais
LD	Cunha (2018)			Práticas sociais
LD	Freitas (2011)	Internet, computador	Compreender, compartilhar	

Fonte: Da autora (2019).

A partir da apresentação do quadro anterior, percebe-se que os conceitos como internet e práticas sociais são recorrentes para muitos autores.

No aspecto técnico de AD (no contexto deste estudo), o indivíduo alfabetizado digitalmente deve ser capaz de manipular recursos tecnológicos (processador de texto, planilhas, apresentações, manipulação de imagens, vídeos, por exemplo), comunicar-se na internet (realizar pesquisas, interagir em redes sociais, fóruns, fazer *upload* e *download*, entre outros). O LD, neste contexto, é entendido como a capacidade de buscar, compreender, reusar, produzir, publicar e/ou compartilhar dados e/ou informações para a comunidade, de forma crítica. Ou seja, o indivíduo está participando da comunidade e exercendo sua plena cidadania. Dessa maneira, segue-se que:

Letramento em ciência e tecnologia para cidadania, então, incorpora conhecimentos e competências que habilitam o cidadão **a tomar decisões pessoais que usem critérios com base em conhecimentos científicos**, como por ex., na decisão sobre compra e utilização de novos equipamentos ou até mesmo sobre um tratamento médico, após ouvir diferentes especialistas (DEL PINO; FRISON, 2011, p. 38, grifo nosso).

Portanto a compreensão sobre LD está em conformidade com as palavras dos autores supracitados, bem como, concorda com o que dizem Barton e Hamilton (2012), quando defendem que há múltiplos letramentos, porque as pessoas, ao utilizarem as tecnologias, têm interesses, necessidades e objetivos diferentes, os quais são extensivos tanto ao uso pessoal quanto ao exercício da profissão.

Importante ressaltar que um indivíduo LD somente poderá atingir este estado se for alfabetizado cientificamente, ou seja, a alfabetização científica é um pré-requisito para a letramento digital.

Até o momento, foram discutidas a Alfabetização Científica (AC) e a Alfabetização Científica e Tecnológica (ACT), esclarecendo a importância no ensino das Ciências e da capacidade do indivíduo para reflexão, posicionamentos críticos e práticas sociais, bem como a Alfabetização Digital (AD) e o Letramento Digital (LD), expondo-se a relevância para a pessoa na sua interação com a sociedade e com as tecnologias. Na seção seguinte, a abordagem será especificamente centrada na Ciência da Computação (CC) e na Alfabetização em Código (ACod).

3.3 Alfabetização em Código (ACod)

O estudo das Ciências, conforme já comentado, desde muito tempo está presente nos currículos escolares. Porém, na Ciência da Computação (CC) o estudo e sua incorporação nos currículos ainda não foi totalmente evidenciado.

Primeiramente, nesta seção, será discutida a CC, explorando se a Computação é ou não considerada uma ciência. Na sequência, são apresentadas discussões sobre a Alfabetização em Código (ACod), conceitos necessários ao entendimento da CC e um diferente enfoque para ACod e Letramento em Código (LCod).

Denning⁴⁵(2012), preocupado com a questão sobre se a CC é uma ciência ou não, organizou o primeiro seminário sobre esse tema em 2012, intitulado: *Ubiquity: The Science in Computer Science*, promovido pela *Association for Computing Machinery* (ACM), sendo um espaço para proposições e debates e de apresentação de propostas de discussões diferentes, em evento anual. Esse mesmo autor pontua que quando a computação se iniciou foi classificada como uma Ciência. Após passado um período de 30 anos foi transformada e entendida como uma Engenharia. Nos últimos 20 anos, retoma sua função científica. Essa discussão tem sido fomentada, recentemente, pelo viés de formação acadêmica das quatro áreas de conhecimento, do acrônimo *Science, Technology, Engineering, and Mathematics* (STEM). Nesse sentido, a CC tem sido entendida como uma Ciência e já está sendo incorporada nos currículos escolares em 13 países europeus e Estados Unidos (VALENTE, 2016).

Por sua vez, Cerf⁴⁶ (2012, p. 3, tradução nossa) questiona “há ciência na ciência da computação?” e apresenta um conjunto de reflexões sobre a ciência, sua importância, seu papel, o que se ensina, e depois conclui que “[...] realmente existe ciência na ciência da computação” (p. 3).

Assim, a partir desses autores, no contexto desta tese, a CC é considerada ciência.

Primeiramente, é necessário definir alguns conceitos específicos desta ciência. Segundo a norma ISO/IEC 2382-1 (Tecnologia da Informação – Vocabulário), da Associação Brasileira de Normas Técnicas (ABNT), *softwares* são “programas, procedimentos, regras e qualquer documentação associada, pertinente à operação de um sistema computacional” (ISO/IEC 2382-1,1998, texto digital), e programa (de computador) é a “unidade sintática que está em conformidade com as regras de uma linguagem de programação particular, composta de

⁴⁵ Peter J. Denning (pjd@nps.edu) é professor de Ciência da Computação e diretor do *Cebrowski Institute for information innovation at the Naval Postgraduate School*, em Monterey, California; é editor da ACM Ubiquity, e ex-presidente da ACM.

⁴⁶ Cerf foi presidente da ACM, em 2012.

declarações, comandos ou instruções necessárias para executar uma certa função ou tarefa, ou ainda solucionar um problema” (texto digital).

No contexto deste trabalho, programa e *software* são utilizados como sinônimos. Ambos estão relacionados com computadores, internet e com a CC. Entende-se que a CC é, entre muitas funções, a geradora de *software* que os usuários consomem, utilizam e compartilham. São os programas desenvolvidos pelos programadores (em um conceito mais genérico denominados cientistas da computação) e que são utilizados pelos usuários, entendendo que o usuário poderá somente consumir (ou interagir com) o programa e apenas estarão disponibilizados os requisitos para os quais o *software* foi planejado e implementado. Explicando de outra forma: os programas são projetados/implementados com determinadas funcionalidades e somente essas estarão acessíveis ao usuário. Não cabe a este último qualquer modificação no funcionamento do *software*. Nesse contexto, os usuários são expectadores. No âmbito deste trabalho não estão sendo analisados os programas ditos de código-aberto (aqueles passíveis de serem alterados para outros fins).

Para que um usuário seja capaz de construir um programa, ou modificar um *software* existente, é imprescindível que possua conhecimentos básicos de algoritmos, de lógica de programação e de, pelo menos, uma linguagem de programação (LP).

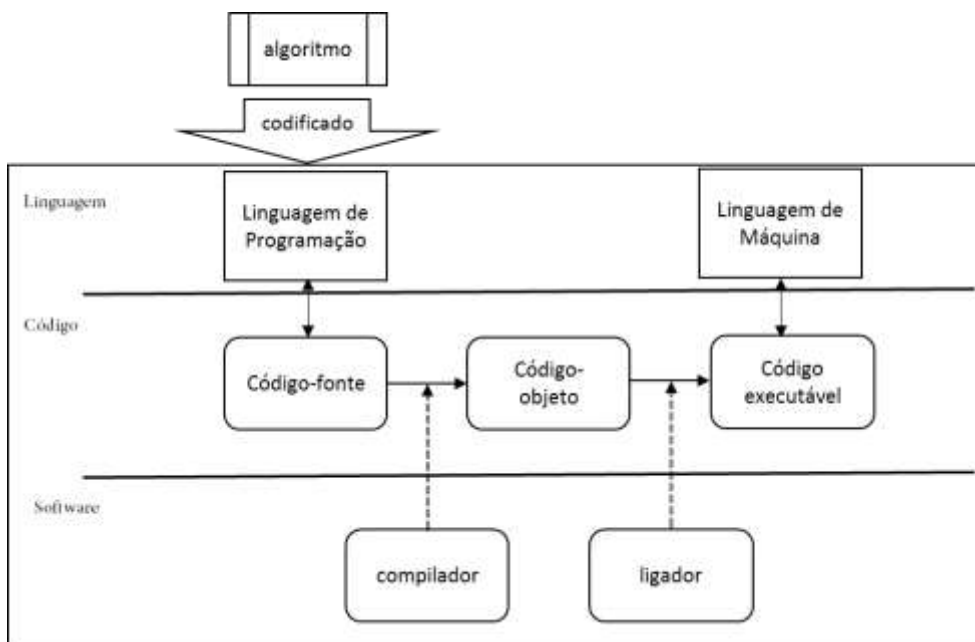
Os algoritmos são um conjunto de passos organizados em uma sequência lógica para resolver um problema. A resolução do problema determina a funcionalidade (ou funcionalidades) que o programa irá incorporar. Já a lógica de programação pode ser entendida como a técnica de encadeamento dos comandos (ou passos) descritos para que um programa funcione.

Por fim, a linguagem de programação (LP) é a forma como se pode interagir com o computador, pois este somente “entende” a linguagem binária, a qual está limitada a zeros e uns e suas infinitas combinações. Por conseguinte, aos humanos fica complicado se comunicar em binário. Por isso, existem as linguagens de programação que são uma forma mais aproximada da linguagem humana. As LPs possuem um conjunto de palavras-chave, também denominados comandos e regras de escrita. A combinação de ambas leva o programador a escrever (codificar) um programa que foi projetado, seja ele um jogo, seja um controlador de mísseis nucleares, por exemplo. Portanto, codificar é aprender a ler e escrever um algoritmo em uma linguagem de programação.

Um programador faz uso de um programa, que se chama “Interface”, para escrever seu código (chamado de código-fonte) em uma determinada LP. Após o programador digitar seu código e informar que está pronto (totalmente escrito seu código-fonte) através da Interface, o

programa será compilado⁴⁷, transformando o código-fonte em código-objeto. Este último será então *linkado* (ligado) pelo programa correspondente, até gerar o programa no formato executável, ou seja, em linguagem de máquina. Este último formato somente será possível se não houver erros nas etapas anteriores. A partir deste processo, e o código livre de erros, o programa é executado. Na Figura 10 pode ser visualizado graficamente o processo descrito.

Figura 10 - Processo que transforma código-fonte em executável



Fonte: Da autora (2018).

Considerando a figura anterior, pode-se apontar que código-fonte, geralmente, é escrito em uma linguagem dita de alto nível, enquanto o código executável está para a linguagem de máquina, também conhecida como de baixo nível. Como exemplo de alto nível tem-se: Linguagem C, C++, *Python*, entre outras.

O foco neste trabalho está em, a partir do desenvolvimento do Pensamento Computacional do indivíduo, que se aproprie das habilidades e competências para programar, ou seja, a partir de um problema, escrever a solução – código-fonte – utilizando uma linguagem de programação específica.

Quando um indivíduo adquire uma linguagem, não aprende apenas a ouvir, mas também a falar. Quando adquire o texto, não aprende somente a ler, mas também a escrever. Com

⁴⁷ Compilar significa que uma linguagem será convertida para uma linguagem de processador, um idioma que o processador entende.

respeito aos computadores, saber utilizar um programa (alfabetizado digital) é diferente de saber construir um programa. Quando os indivíduos não são alfabetizados em código, devem aceitar os *softwares* que utilizam com quaisquer limitações que seus criadores tenham embutido neles.

A partir da figura do programador, e em específico na área da CC, surge a expressão “*code literacy*”, traduzida por ‘alfabetizado em código’. Ou, em espanhol, “*codigoalfabetización*”, definido como sendo o “processo de ensino-aprendizagem da leitura e escrita de linguagens de computadores e outras máquinas e de pensar computacionalmente” (ROMÁN GONZÁLEZ, 2014, p. 136, tradução nossa).

Para Belshaw (2013, p. 93, tradução nossa), *code literacy* (alfabetização em código) é “a capacidade de ler e escrever uma linguagem de máquina e pensar computacionalmente [...]”. Esse autor defende que a ACod é necessária, porém não é suficiente, no sentido amplo de alfabetização digital, pois argumenta que o indivíduo precisa muito mais conhecimentos para se tornar membro efetivo da sociedade. Em virtude disso, apresenta o conceito de *web literacy*, que pode ser entendido como “[...] as habilidades e competências necessárias para ler, escrever e participar efetivamente online” (BELSHAW, 2014a, texto digital).

Em 2013, foi lançado o *Web Literacy Standard* (Padrão para Alfabetização na Web), porém não foi muito aceito pela comunidade de desenvolvedores, pela questão do termo “padrão”, o qual não havia sido apresentado e discutido pela comunidade. Uma nova proposta surgiu em 2014 (BELSHAW, 2014b) com o nome de ‘*Web Literacy Map*’ (Mapa da Alfabetização na Web). Neste mapa, que é apresentado na Figura 11, está sinalizado que há ou pode haver codificação – ou seja, escrita de código-fonte. Porém, essa programação, através de tecnologias *Web* e/ou LPs, são especificamente para produtos *Web*.

Figura 11- Mapa Digital



Fonte: Adaptado pela autora com base em Belshaw (2014b).

Retomando a ACod, Dufva (2013), em seu livro “*Code Literacy. Understanding the programmed world*”, apresenta o conceito de ACod como três etapas distintas. A primeira trata de como os programas são construídos e o que são as linguagens de programação. A segunda etapa aborda os programas e suas limitações e estuda as tecnologias de forma geral. E na última etapa, estuda o desenvolvimento propriamente dos programas. Esse mesmo autor complementa que o indivíduo precisa não só desenvolver habilidades de ACod, pois esta “certamente pode nos beneficiar individualmente a fim de nos tornarmos melhores usuários de tecnologia” (DUFVA, 2013, p. 83, tradução nossa), como também adquirir autocontrole no mundo digital.

A ACod, para Betts (2011, texto digital, tradução nossa), é entendida como sendo “a habilidade para identificar, entender, interpretar, criar, comunicar e usar regras que configuram e reconfiguram informações para serem utilizadas na criação de novas informações”. Pode-se

perceber que esse autor apresenta sua definição a partir do entendimento e definição proposta pela UNESCO e acrescenta novas informações.

No entendimento deste trabalho, a definição não está de acordo com a questão do “código”. Primeiramente porque este não é mencionado e também quanto à questão da palavra “informação”. O entendimento de ACod está no foco de escrita e leitura de código e, mais especificamente, em código-fonte. Esse é entendimento como sendo a escrita de um programa obedecendo às regras e recomendações da LP.

Para Prensky (2008, texto digital, tradução nossa), a habilidade de codificar ou programar é definida como: “a capacidade de usar a tecnologia digital para fazer o que for possível, dentro do possível, de acordo com as suas necessidades, finalidades e vontade”. Em inglês, o autor utiliza a expressão ‘*programming literacy*’. Esse pesquisador enfatiza o quão importante é para o indivíduo dominar vários tipos de interações entre homens-máquina, as quais denomina de “novas alfabetizações”.

Concordando com Prensky (2008), Resnick (2015) defende que a codificação é uma forma diferente de escrita, sendo uma nova alfabetização, que capacita as pessoas a mostrarem o que acontece em suas mentes. Ademais, defende que a pessoa alfabetizada em código pensa de forma sistemática e como consequência altera sua forma de agir e sua organização pessoal, nas tarefas diárias. O resultado do seu trabalho é o *Scratch*⁴⁸ (RESNICK *et al.*, 2009), que é uma plataforma *online* para crianças apreenderem programação, ou seja, desenvolverem as habilidades e competências para ACod. Esse autor complementa que:

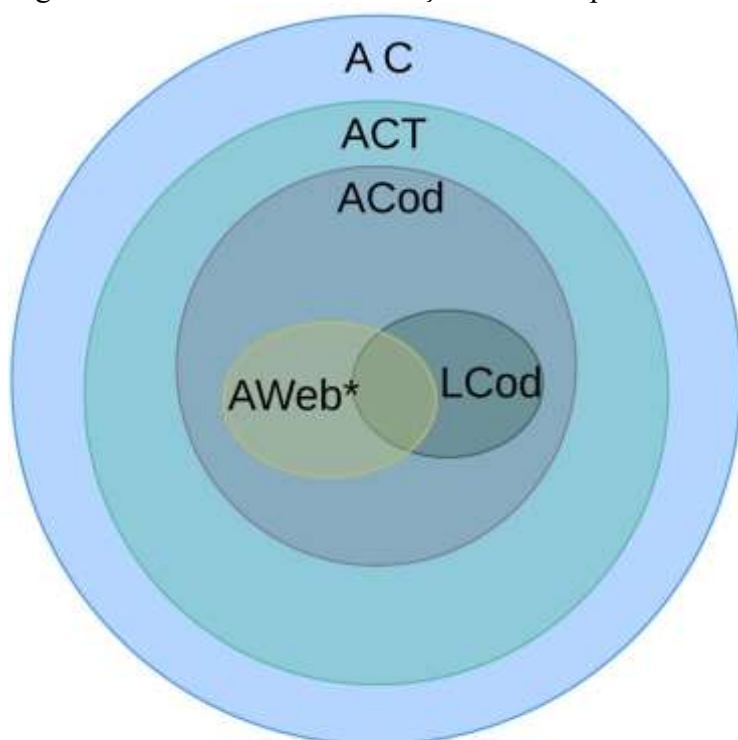
Embora elas (pessoas) interajam com mídia digital o tempo todo, poucos conseguem criar seus próprios jogos, animações ou simulações. É como se eles pudessem ‘ler’ mas não ‘escrever’... a fluência digital requer não apenas a capacidade de conversar, navegar e interagir, mas também a capacidade de projetar, criar e inventar com novas mídias... **precisa aprender algum tipo de programação.** A capacidade de programar oferece importantes benefícios [...] (RESNICK *et al.*, 2009, p. 62, tradução e grifo nosso).

A partir da leitura e do entendimento dos autores citados com relação à ACod, perceberam-se diferentes palavras para descrever o mesmo significado para os termos.

As diferenças de alfabetização e letramento científico e tecnológico, apresentadas por vários autores já mencionados e mais especificamente no contexto de “tecnológico”, o entendimento no âmbito desta pesquisa é apresentado a seguir, e pode ser acompanhado na Figura 12.

⁴⁸ Disponível em: <https://scratch.mit.edu/>. Acesso em: 12 maio 2018.

Figura 12 - Níveis de Alfabetização e suas dependências



* AWeb representa quaisquer tipos de alfabetização.

Fonte: Da autora (2018).

Observando o diagrama da Figura 12, um indivíduo alfabetizado cientificamente é aquele que sabe ler e escrever em uma determinada linguagem. É pré-requisito para o indivíduo AC e ACT, pois precisa saber ler, escrever e fazer uma interpretação da realidade e da sociedade na qual está inserido. Então, corresponde a estar habilitado para manipular as TDIC e recursos digitais. Dentro dessa perspectiva, podem haver vários tipos de alfabetizações como, por exemplo, a alfabetização para Web.

Continuando, para ser um indivíduo ACod, necessariamente é preciso ser ACT. O conceito utilizado sobre ACod segue, parcialmente, Román González (2014, p. 5, tradução nossa), o qual define como “ o processo de ensino-aprendizagem da leitura e escrita com uma linguagem de programação”. Por que por somente uma LP? Entende-se que quando uma pessoa é alfabetizada, aprende a ler e a escrever em uma determinada linguagem, por exemplo, em português. Saber a língua portuguesa não habilita a ler e a escrever em espanhol, por exemplo, ou em qualquer outro idioma. Logo, uma pessoa (no caso, programador) sabe ler e escrever código (mais especificamente o código-fonte deverá ser escrito em uma e somente uma LP).

Considerando que para escrever um código (independente da LP) é preciso conhecer a sintaxe⁴⁹ e a semântica da linguagem⁵⁰ (embora esta segunda esteja vinculada com o desenvolvido do PC), além de considerar que as sintaxes são diferentes, entende-se que uma pessoa é ACod quando domina apenas **uma** LP – pois conhece e aplica sua sintaxe (CORDENONZI *et al.*, 2020).

Semanticamente as LPs podem apresentar semelhanças, desde que obedeçam ao mesmo paradigma de programação⁵¹. Do mesmo modo, para escrever um código em uma determinada LP precisa-se de uma ferramenta que faça a interface entre o programador e o computador. Entre muitas ferramentas há o *Integreted Development Environment (IDE)*, ou ambiente de desenvolvimento integrado, que é o programa que integra as várias ferramentas necessárias para o desenvolvimento de *softwares*, anteriormente denominado de Interface. Mas não é a LP.

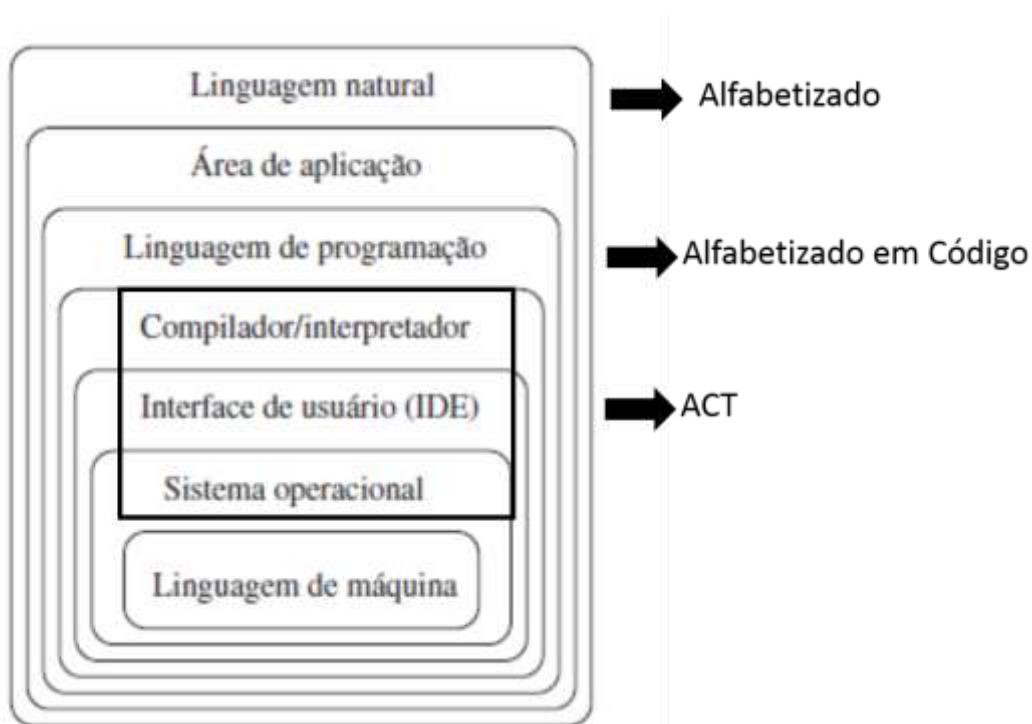
Para usar, organizar, operar, manipular com a ACod, também são necessários conhecimentos sobre os sistemas operacionais e sua organização física (arquivos, pastas, entre outros), os tipos de meios digitais nos quais será utilizado seu programa (computadores pessoais, dispositivos móveis, por exemplo), redes de dados e internet. Enfim, para se ter um indivíduo ACT, ele precisa estar habilitado em, pelo menos, alguns tipos de alfabetização digital, de acordo com a sua necessidade e interesses. A Figura 13 mostra a relação entre as ferramentas e os tipos de alfabetização.

⁴⁹ “A sintaxe de uma linguagem descreve o que constitui um programa estruturalmente correto” (TUCKER; NOONAN, 2010, p. 2).

⁵⁰ “O significado de um programa é definido pela sua semântica, ou seja, quando um programa é executado, o efeito de cada comando sobre os valores das variáveis no programa é dado pela semântica da linguagem” (TUCKER; NOONAN, 2010, p. 3).

⁵¹ Definido “como um padrão de pensamento que guia um conjunto de atividades relacionadas. Um paradigma de programação é um padrão de resolução de problemas que se relaciona a um determinado gênero de programas e linguagens” (TUCKER; NOONAN, 2010, p. 3).

Figura 13 - Ferramentas e Alfabetizações



Fonte: Adaptado pela autora com base em Tucker e Noonan (2010, p. 13).

A partir da análise da Figura 13, pode-se perceber o entrelaçamento e as dependências dos diferentes conceitos e/ou *softwares* envolvidos para construir um programa, preconizados pelos autores, ou seja, ao desenvolvedor de *software* são exigidos diferentes tipos de alfabetizações e todos são necessários.

Portanto, uma pessoa que desenvolve seu PC, sua capacidade de resolver problemas, desenvolve habilidades e competências para ler, escrever e interpretar código-fonte em uma linguagem de programação, é considerada como ACod (CORDENONZI *et al.*, 2020).

Quando extrapola seus conhecimentos para mais de uma LP é então denominada de Letramento em Código (LCod) (CORDENONZI *et al.*, 2020). É essencial elucidar que neste contexto as LPs são independentes de seus paradigmas de programação e seus tipos (procedurais, orientadas a eventos, entre outros) ou nível (alto ou baixo).

É importante pontuar que neste estudo não foi abordada a questão da complexidade de desenvolvimento de código. Ou seja, até o presente, a ACod ou LCod não estão considerando a dificuldade da resolução de problema e a sua correspondente codificação.

Em síntese, resumindo os conceitos principais aqui apresentados, tem-se que:

- Um indivíduo **Alfabetizado em Código** (Acod) é aquele capaz de ler, interpretar e escrever um código-fonte e gerar um código executável (programa).

- Um indivíduo **Letrado em Código** (LCod) é um ACod extrapolando suas habilidades e competências para codificar em diferentes linguagens de programação (CORDENONZI *et al.*, 2020, p. 150).

Por fim, as considerações para concluir este capítulo.

3.4 Considerações e contribuição

Ao longo deste capítulo, pode-se perceber a importância do conhecimento científico na área das Ciências em geral, bem como seu devido reconhecimento para ser ensinado nas escolas. No decorrer dos tempos, a Alfabetização Científica (AC) foi sendo incorporada e auxiliada pelas ferramentas digitais e tecnológicas. Novos acrônimos foram sendo propostos, como, por exemplo, ACT, de Alfabetização Científica e Tecnologia.

Não se pode desconsiderar a presença da tecnologia no dia a dia; logo, a inclusão digital tornou-se uma necessidade e preocupação para entidades educadoras e governamentais. A Alfabetização Digital (AD) tornou-se imprescindível para os cidadãos.

Mais recentemente e especificamente na Ciência da Computação, surgindo da necessidade do mercado de trabalho e/ou de melhorar o processo de resolução de problemas (tanto de programação como de problemas do cotidiano), tornou-se igualmente necessária a Alfabetização em Código (ACod). Ou, ainda de forma mais abrangente, o Letramento em Código (LCod).

O entendimento proposto por Rushkoff (2012) acerca de ACod defende que essa alfabetização se inicie nos primeiros anos escolares, pois é um requisito importante para a interação do indivíduo no mundo digital. A partir do entendimento apresentado pelo autor sobre a ACod, esta tese contribuiu para a conceituação de Alfabetização em Código (ACod) e Letramento em Código (LCod). Também contribuiu para diferenciar a Alfabetização Digital (AD) e Letramento Digital (LD).

4 PROCEDIMENTOS METODOLÓGICOS

A intenção deste capítulo é apresentar a descrição dos procedimentos metodológicos aplicados, desde a definição até a conclusão desta tese, perpassando todo o processo experienciado para responder aos objetivos definidos no início deste trabalho. Este ponto não é o final, pois se acredita que a pesquisa e sua aplicabilidade poderão prosseguir.

Para o melhor entendimento, o texto está organizado em duas seções. Na primeira, apresenta-se a caracterização da pesquisa, ou seja, os métodos, técnicas, instrumentos utilizados, população envolvida, entre outros. Já na seção seguinte, o desenvolvimento deste trabalho juntamente com as propostas e constructos metodológicos estão detalhados.

4.1 Caracterização da pesquisa

Uma pesquisa tem por objetivo encontrar respostas a uma pergunta ou problema (CHEMIN, 2020), e/ou buscar um dado ou uma informação que ainda não se conhece ou não se domina, podendo ser entendida como um “ procedimento racional e sistemático que tem como objetivo proporcionar respostas aos problemas que são propostos” (GIL, 2002, p. 17). Ainda nas palavras deste autor, “é descobrir respostas para problemas mediante o emprego de procedimentos científicos” (GIL, 2008, p. 26). Portanto, a partir do uso de metodologias e técnicas, o pesquisador desenvolve o que denomina de pesquisa científica. Para complementar, Minayo (2002, p. 25) define que a pesquisa é o uso de “ uma linguagem fundada em conceitos, proposições, métodos e técnicas, linguagem esta que se constrói com um ritmo próprio e particular”.

Os procedimentos metodológicos, ou metodologia, são entendidos como “o caminho do pensamento e a prática exercida na abordagem da realidade” (MINAYO *et al.*, 2002, p. 16),

cujo caminho vai exigir do pesquisador três olhares e discussões: o primeiro é sobre as teorias consideradas; o segundo, sobre as técnicas utilizadas no desenvolvimento, e, o último, sobre a interpretação dos resultados, ou seja, “o potencial criativo do pesquisador” (p. 17).

A fim de retomar os princípios essenciais deste trabalho, o problema norteador da pesquisa é: “Como acontece⁵² o processo de ensino e aprendizagem sobre PC nos sujeitos (binacionais) que estão no ensino pós-médio?” e as hipóteses formuladas para serem confirmadas ou não são:

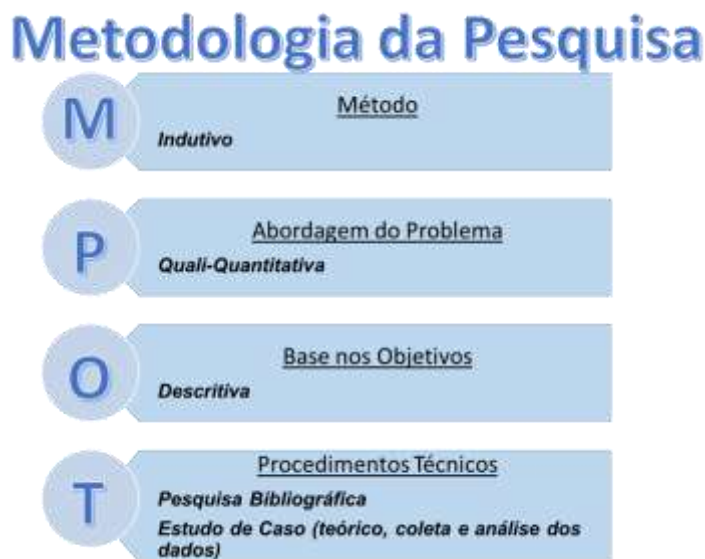
- a) O sujeito pode desenvolver as habilidades (Acod) de PC.
- b) O sujeito pode desenvolver competências (Lcod) de PC.
- c) O sujeito pode desenvolver um projeto de *software* utilizando PC.
- d) Há diferenças no resultado de formação do PC entre os sujeitos com formações diferentes de ensino, ou seja, entre brasileiros e uruguaios.

O início de uma pesquisa científica se dá quando existe um problema possível de ser solucionado (GIL, 2002; LAKATOS; MARCONI, 2003; MINAYO *et al.*, 2002). Assim, a partir da pergunta descritiva do problema a ser pesquisado, as hipóteses foram formuladas e a pesquisa tomou forma para ser levada adiante. Acrescenta Triviños (2015) que o problema de pesquisa é um cenário ou contexto que precisa ser esclarecido, fato este claramente já mitigado na justificativa da aprendizagem significativa aliada à andragogia, ainda somados aos escassos trabalhos sobre PC em adultos. Além disso, esse mesmo autor sugere que “o assunto deve surgir da prática cotidiana que o pesquisador realiza como profissional” (TRIVIÑOS, 2015, p 93); portanto, atendida com os vinte anos de prática docente (descrita na seção 1.4). Na Figura 14, está resumida a organização desta pesquisa e, após, a explanação de cada uma das escolhas realizadas.

Este trabalho foi realizado seguindo o método indutivo, que é entendido como sendo aquele que “parte do particular e coloca a generalização como um produto posterior do trabalho de coleta de dados particulares” (GIL, 2008, p. 10). Esse autor propõe ao pesquisador que inicie seu trabalho a “partir de fatos ou fenômenos cujas causas deseja conhecer” (p.10), ou seja, de fatos concretos com a intenção de compará-los entre si. Ainda, argumenta que a observação é um “procedimento indispensável para atingir o conhecimento científico” (p.11). Ademais, Lakatos e Marconi (2003, p. 92) confirmam que o método “indutivo tem o objetivo de ampliar o alcance dos conhecimentos”.

⁵² Como ocorre: no sentido de o adulto conseguir desenvolver, primeiramente, o interesse (andragogia) e perceber sua necessidade da aprendizagem, medindo o interesse em resolver o problema, ou seja, sua habilidade, e depois sua competência (resolução inovadora no sentido de criação de uma solução própria para o problema).

Figura 14 - Metodologias utilizadas



Fonte: Da autora (2020).

Quanto à abordagem do problema, esta pesquisa é quali-quantitativa. De acordo com Triviños (2015), nas pesquisas qualitativas, tem-se o ambiente natural como origem para coletar os dados; sendo que o pesquisador e sua interação são considerados instrumentos-chave. Ainda, nesse tipo de abordagem, a tendência do pesquisador é descrever os dados a partir de uma análise indutiva. Importante salientar que os pesquisadores qualitativos não estão preocupados somente com os resultados esperados e/ou encontrados ou somente com o produto, mas também com o todo o processo percorrido e a atribuição de significado.

Já as pesquisas quantitativas estão centradas na objetividade, em que a análise dos dados está centrada nos números (FONSECA, 2002) e nos seus significados. Gatti (2004, p. 13) confirma que estes significados “podem ser muito úteis na compreensão de diversos problemas educacionais”.

Com a combinação de ambas as abordagens, consegue-se recolher mais dados e informações “muito úteis na compreensão de diversos problemas educacionais” (GATTI, 2004, p. 13), além de “que permitem interpretações de diversas naturezas” (GATTI, 2012, p. 31) e torna-se possível dar sentido e idealizar interpretações a partir deles. Entende-se que não são antagônicas, e, sim, complementares, pois permitem ao pesquisador olhares distintos (PEREIRA; ORTIGÃO, 2016).

Já com base nos objetivos baseia-se no método descritivo. A pesquisa descritiva utiliza questionários e observações para coletar dados, embora não exclua outros tipos. Conforme Gil (2008, p. 117):

A manipulação qualitativa dos dados durante a análise é uma atividade eclética; não há uma única maneira de fazê-la. Embora se reconheça a importância de um arcabouço metodológico sólido, não se pode dispensar a criatividade do pesquisador. Cabe-lhe muitas vezes desenvolver a sua própria metodologia.

Tal qual a afirmação anterior, corroborada por Triviños (2015) e Gil (2002), a maioria das pesquisas no campo da educação são desenvolvidas de forma descritiva. Esses autores entendem que neste tipo de pesquisa o pesquisador está procurando conhecer profundamente a comunidade (sujeitos). Triviños (2015, p 15) acrescenta que um dos procedimentos técnicos mais utilizados é o estudo de caso, que define como sendo “[...] estudos que têm por objetivo aprofundarem a descrição da determinada realidade”. Por isso, pode-se afirmar que esta investigação está alinhada com as proposições descritas.

Com base nos procedimentos técnicos utilizados, em um primeiro momento, a respeito da leitura em diversas fontes de dados, Gil (2002) salienta que este tipo de pesquisa é importante e, portanto, utiliza-se de diferentes autores que abordam o mesmo assunto em documentos reconhecidos pela comunidade científica. Ainda, este mesmo autor complementa que “a principal vantagem da pesquisa bibliográfica reside no fato de permitir ao investigador a cobertura de uma gama de fenômenos muito mais ampla do que aquela que poderia pesquisar diretamente” (GIL, 2002, p. 45). Sendo assim, a partir das suas recomendações, foram consultados diversos trabalhos com diferentes pesquisadores, com a intenção de entender seus posicionamentos e produzir novos conhecimentos.

Ainda no campo dos procedimentos técnicos, foi utilizado o estudo de caso. Do ponto de vista de Gil (2002, p. 54), pode ser definido como “estudo profundo e exaustivo de um ou poucos objetos, de maneira que permita seu amplo e detalhado conhecimento”. Já Triviños (2015) define que o estudo de caso é uma unidade, a qual deve ser estudada profundamente e aconselha que os dados podem ser apresentados de forma qualitativa. Por sua vez, Yin (2001) propõe o estudo de caso como um método rigoroso a ser utilizado por pesquisadores, no qual enfatiza muito o planejamento e análise dos dados.

Para apoiar a decisão de utilizar o estudo de caso, buscaram-se as palavras de Yin (2001, p. 26): “Para o estudo de caso, isso ocorre quando: faz-se uma questão do tipo ‘como’ ou ‘por que’ sobre um conjunto contemporâneo de acontecimentos sobre o qual o pesquisador tem pouco ou nenhum controle”.

A investigação do estudo de caso deve basear-se no maior número de fontes de evidências possíveis com o intuito de convergir na triangulação dos dados, resultando no prévio desenvolvimento e construção teórica que deverão conduzir as etapas de coleta e análise dos dados (YIN, 2001). Porém, salienta que, para aumentar a confiabilidade, o pesquisador deverá

manter o encadeamento dessas evidências. Esse mesmo autor argumenta que, quanto ao tempo, este método “mostra que é possível a realização de estudos de caso em períodos mais curtos e com resultados passíveis de confirmação por outros estudos” (YIN, 2001, p. 55).

Para a aplicação dos estudos de casos, é necessário que o pesquisador desenvolva um protocolo, cujo documento deverá conter os procedimentos, as regras e as demais informações que se julgar importante para conduzir a pesquisa, principalmente se forem executados múltiplos estudos de casos. Como consequência desse instrumento, aumenta-se a confiabilidade da pesquisa (YIN, 2001; GIL, 2002).

Com relação à quantidade de número de casos:

O procedimento mais adequado para esse fim consiste no adição progressivo de novos casos, até o instante em que se alcança a ‘saturação teórica’, isto é, quando o incremento de novas observações não conduz a um aumento significativo de informações [...] que inseridas em diferentes contextos, concorrendo para a elaboração de uma pesquisa de melhor qualidade (GIL, 2002, p. 140).

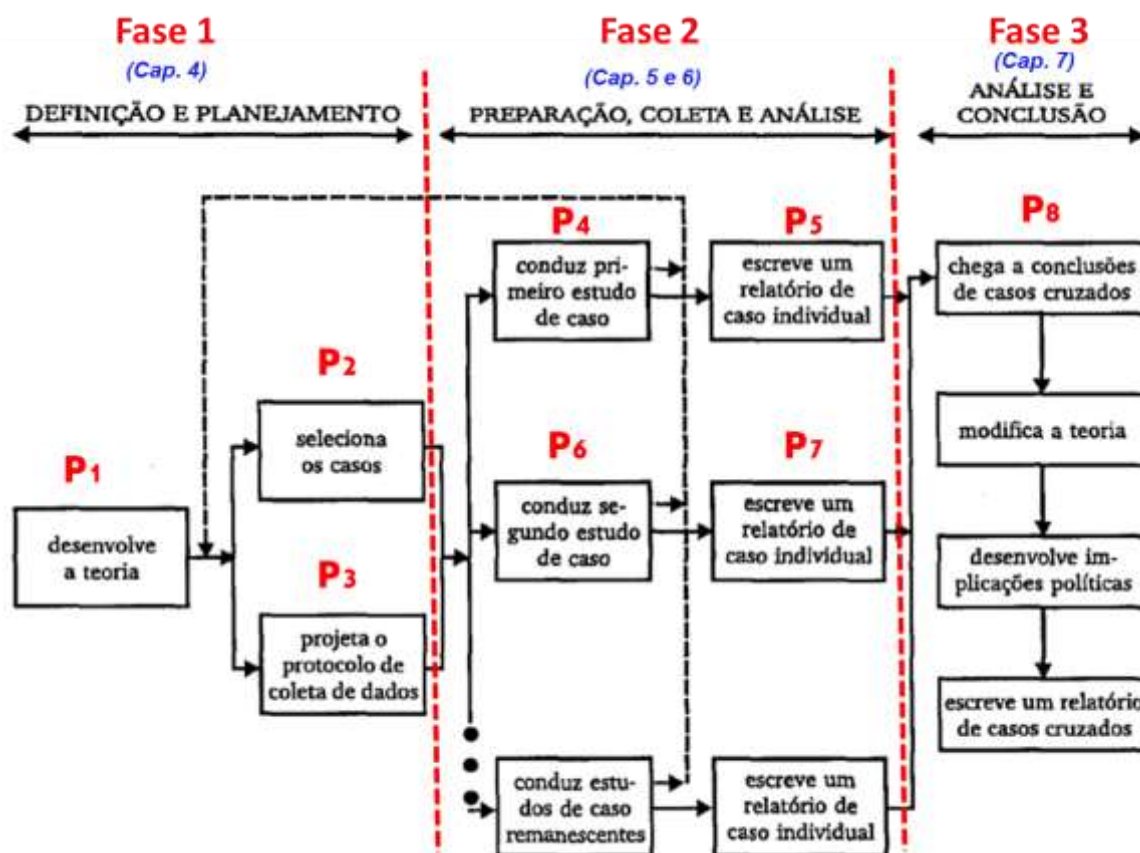
A coleta dos dados pode ser realizada por meio de um caso piloto, sugere Yin (2001, p. 100):

O estudo de caso piloto auxilia os pesquisadores na hora de aprimorar os planos para a coleta de dados tanto em relação ao conteúdo dos dados quanto aos procedimentos que devem ser seguidos. Nesse sentido, é importante observar que um teste-piloto não é um pré-teste. O caso-piloto é utilizado de uma maneira mais formativa, ajudando o pesquisador a desenvolver o alinhamento relevante das questões.

A análise dos dados deve ser desenvolvida para verificar se os objetivos da pesquisa foram atendidos e para relacionar os dados com a finalidade de confirmar ou rejeitar a(s) hipótese(s) ou os seus pressupostos (LAKATOS; MARCONI, 2003). A análise será qualitativa, e isso significa que não existem receitas, fórmulas ou algoritmos para orientar o pesquisador, porém deve ser sistemática, reflexiva, comparativa, eclética e o resultado deverá “ser uma síntese em mais alto nível” (GIL, 2008, p. 177). Para Lakatos e Marconi (2003), a análise quantitativa dos dados precede a interpretação significativa mais ampla para responder as questões levantadas e aproximando-as a outros conhecimentos. Como definido, a pesquisa qualitativa trabalha com a totalidade dos significados das ações, das relações humanas e sua interpretação, de forma descritiva, explanando e explicando os achados da pesquisa (MINAYO *et al.*, 2002).

Na Figura 15, o método do estudo de caso, proposto por Yin (2001), está representado.

Figura 15 - Método de estudo de caso



Fonte: Adaptado pela autora com base em Yin (2001, p. 73)⁵³.

Yin (2001) propõe uma organização para ser seguida pelo pesquisador ao utilizar o método de estudo de caso. Inicialmente, organiza em três momentos diferentes que são: Definição e Planejamento, Preparação, Coleta e Análise dos Dados e, por fim, Análise e Conclusão, os quais podem ser identificados na Figura 15. Nesta tese são chamados de Fase.

Na Fase 1, devem ser executados três passos, sendo que no primeiro cabe ao pesquisador desenvolver sua teoria, seus instrumentos de pesquisa, o protocolo, entre outros. Os passos 2 e 3 podem ocorrer de forma paralela, sendo que no Passo 2 devem-se definir os casos a serem executados. Já no passo 3, os instrumentos avaliativos da pesquisa são definidos. Ao completar esses passos, o pesquisador poderá ingressar na Fase 2, executando os passos 4 e 5, que são a realização do estudo de caso (sendo o primeiro, o caso piloto) e escrevendo o relatório deste caso (passo 5). Sucessivamente se repete a Fase 2 para tantos casos quantos forem necessários. Quando terminar esta fase, o investigador passa para a Fase 3 – análise e conclusão – na qual faz o cruzamento dos resultados dos estudos realizados, a triangulação dos dados e, a partir daí,

⁵³ As modificações feitas do modelo original estão nas cores vermelho e azul.

pode modificar a teoria e desenvolver as implicações políticas (ambas opcionais), findando o processo através do relatório conclusivo.

Nas seções seguintes, será apresentado o desenvolvimento dos procedimentos metodológicos desta tese, ou seja, os Passos 1, 2 e 3 da Fase 1. Os passos P4 e P5 (que corresponde ao caso piloto⁵⁴) estão descritos no Capítulo 5. Os passos P6 e P7 estão narrados no Capítulo 6 e, por fim, no Capítulo 7, a Fase 3 (análise quantitativa e conclusão) está retratada.

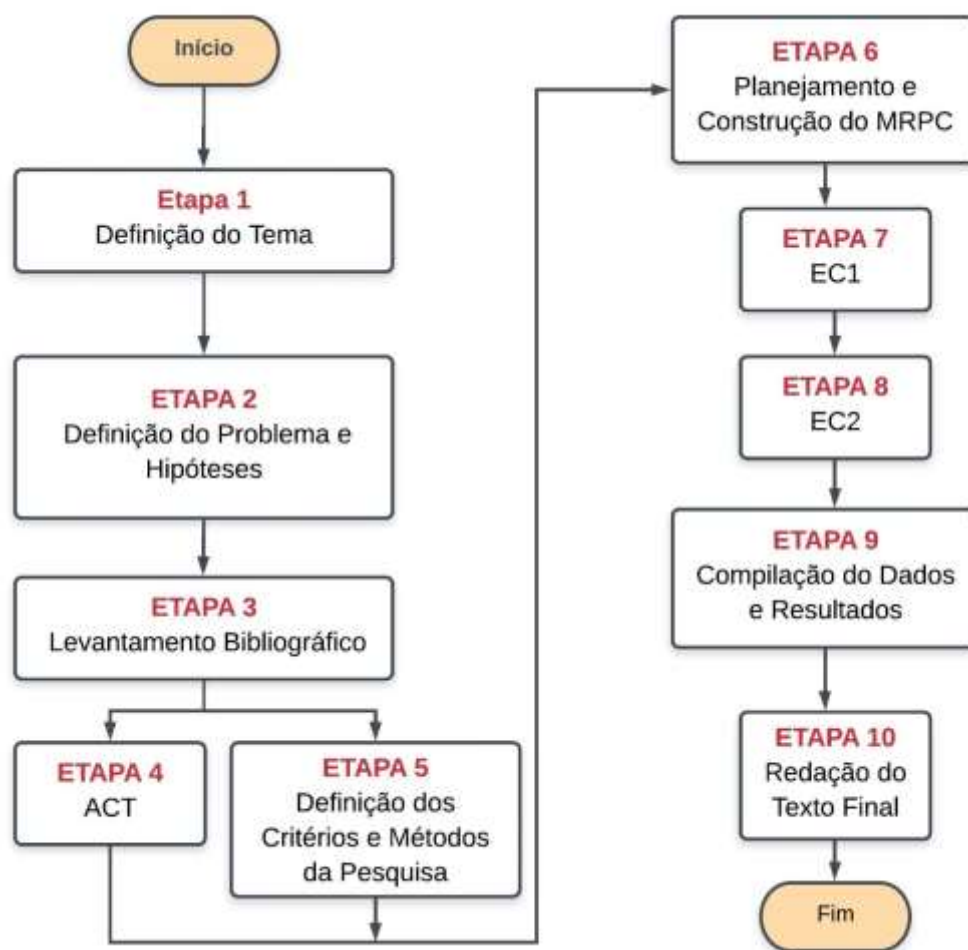
4.2 Desenvolvimento da pesquisa

O desenvolvimento desta seção tem por objetivo responder à pergunta norteadora desta pesquisa, por intermédio da busca de como acontecem os processos de ensino e aprendizagem sobre pensamento computacional (PC) nos sujeitos (binacionais), os quais estão matriculados em um curso de nível de ensino pós-médio. Ou seja, como o adulto desperta, primeiramente, o interesse (andragogia) e a percepção da necessidade da aprendizagem no que tange à habilidade e competência da alfabetização em código. Ademais, nesta seção o objetivo principal e os específicos são detalhados.

Para tal, muitas etapas foram desenvolvidas e muitos instrumentos foram planejados, construídos e validados. Estas etapas, que representam as atividades realizadas, podem ser visualizadas na Figura 16.

⁵⁴ No contexto desta tese, caso piloto e estudo de caso 1 (EC1) têm o mesmo significado.

Figura 16 - Etapas da pesquisa



Fonte: Da autora (2019).

Retomando-se a afirmação de Gil (2008) de que a pesquisa inicia com a definição de um tema, a primeira etapa deste trabalho foi o tema/objeto de estudo: “O Pensamento Computacional (PC) no adulto”. Na etapa seguinte, definiu-se o problema e as hipóteses. Para isso, algumas leituras foram realizadas, a fim de situar a autora no contexto atual sobre o PC, avaliação de PC, entre outros, já abordados no Capítulo 2. Ao término desta etapa, e em alguns momentos de forma concomitante, a Etapa 3 começou a ser detalhada. Nesta etapa, talvez uma das mais longas do trabalho, muitas leituras foram realizadas, com o propósito de levantar o estado da arte dos vários conteúdos, assuntos, conceitos, ferramentas que estão envolvidos na elaboração desta pesquisa (relatados no Capítulo 2). Em virtude dessas leituras, seguiu-se em dois caminhos paralelos em muitos momentos. Na Etapa 4, dados e informações sobre ACT foram coletados e geraram a primeira contribuição desta tese para a comunidade científica, com a finalidade de diferenciar os sujeitos enquanto seu conhecimento em código. Em suma,

definiu-se alfabetização em código e letramento em código. Os requisitos desta classificação estão especificados na próxima seção.

Os critérios para o desenvolvimento da pesquisa científica (conforme pode ser visualizado na Figura 14) foram delineados na Etapa 5. Já na Etapa 6, o método desenvolvido, os instrumentos de pesquisa e de avaliação e demais documentos (protocolo) elaborados foram desenhados e estão descritos nas seções seguintes. Além desses, os documentos exigidos pela Resolução N°466/2012 do Conselho Nacional de Saúde, como o Termo de Consentimento Livre e Esclarecido (TCLE) e a Carta de Anuência das Instituições envolvidas nos estudos de casos encontram-se, respectivamente, nos Apêndices C, D e E. Outro procedimento realizado, já com todos os documentos finalizados, foi a solicitação de registro na Pró-Reitora de Pesquisa, Inovação e Pós-Graduação do IFSUL, que está catalogado sob o número PD00181118/156. Esta pesquisa está registrada no Comitê de Ética sob o número 3.551.029.

Na sequência, os estudos de casos foram realizados (Etapa 7 e 8), analisados (Etapa 9) e descritos (Etapa 10).

Os sujeitos da pesquisa são os alunos matriculados em cursos binacionais. O primeiro estudo de caso (EC1), ou caso piloto, foi desenvolvido com os alunos do curso *Tecnico Terciario en Logistica* e o EC2 com os estudantes do curso superior *Análise de Desenvolvimento de Sistemas*.

A partir do levantamento dos muitos objetos envolvidos no tema deste trabalho, e considerando os seus objetivos, descreve-se minuciosamente um curso denominado de “Eu Programa 1.0!”, com a intenção de responder à questão de pesquisa. Todo o constructo andragógico do curso está apresentado nas Seções 4.2.2 e 4.2.3, e os seus instrumentos avaliativos na seção que segue. Como uma das primeiras atividades, um conjunto de perguntas (questionário denominado de ‘Conhecendo Você’) é proposto aos alunos para que o respondam. A sua finalidade é conhecer os sujeitos participantes em diferentes aspectos: idade (para confirmar se são adultos), nacionalidade (para verificar se são brasileiros ou uruguaios), trabalhadores e se contribuem com a renda familiar, ou seja, se estão no mercado de trabalho, e, por fim, qual a expectativa com respeito ao curso (disposição para aprender, aplicação prática no seu cotidiano). Outras questões que também compõem este questionário estão relacionadas aos seus conhecimentos sobre: se já sabe programar (prática), sobre conceitos como algoritmos e comandos básicos de programação. Consideram-se esses aspectos muito importantes para que o professor entenda os conhecimentos prévios (subsunçores) de cada aluno, a fim de adequar a melhor forma de desenvolver o curso. Essas respostas são o ponto inicial para traçar o perfil do desenvolvimento do PC e a evolução de cada indivíduo participante.

A seguir a definição das habilidades no âmbito deste trabalho.

4.2.1 Proposição das habilidades

Retomam-se algumas ideias apresentadas anteriormente, principalmente no referencial teórico, e, em virtude dos muitos entendimentos sobre PC, este trabalho segue o conceito proposto por Wing (2017). Essa estudiosa argumenta que humanos podem computar, em contraste com a ideia de que o PC pode ser utilizado sem computador e complementa que o PC não é somente resolução de problemas, mas também a formulação de problemas.

Retomando o Quadro 1, no qual foi apresentado um resumo dos conceitos e habilidades necessários e/ou importantes para desenvolver o PC, as palavras “resolução de problemas”, “algoritmos” e “abstração” foram as mais citadas, sendo esta última considerada a mais importante, segundo os autores pesquisados.

Na literatura há muitas divergências sobre a definição de habilidades e competências (FROTA, 2017). Em geral, as habilidades são consideradas como algo menos amplo do que as competências. Assim, a competência estaria constituída por várias habilidades. Entretanto, uma habilidade não "pertence" a determinada competência, uma vez que uma mesma habilidade pode contribuir para competências diferentes (GARCIA, 2005).

Habilidade é o saber-fazer, ou o conjunto de saberes, as capacidades, as informações, os recursos cognitivos, o conhecimento. Seguindo, então, a competência é a faculdade de mobilizar o saber-fazer a fim de “solucionar com pertinência e eficácia uma série de situações” (PERRENOUD, 1999a, p. 30). Nas palavras de Ramos (2001, p. 49), “quando as capacidades [habilidades] são colocadas a serviço da ação, competências são desenvolvidas e se tornam aprendizados interiorizados pelos sujeitos”. O conceito de habilidade, para Araújo (2019, p. 28), é declarado como sendo “a faculdade de executar uma ação destinada a cumprir a meta”, já a competência são as ações para atingir um objetivo.

Conforme esclarece Garcia (2005, p. 5):

[...] uma pessoa, por exemplo, que tenha uma boa expressão verbal (considerando que isso seja uma habilidade) pode-se utilizar dela para ser um bom professor, um radialista, um advogado, ou mesmo um demagogo. Em cada caso, essa habilidade estará compondo competências diferentes.

Portanto, no âmbito deste trabalho, o entendimento de que as habilidades estão associadas ao saber-fazer, ou seja, pode ser uma ação física ou mental que aponta para uma capacidade adquirida (MORETTO, 1999). Já as competências são um conjunto ou um feixe de

habilidades desenvolvidas uniformemente (MORETTO, 1999; PERRENOUD *et al.*, 2002). Em outras palavras, é a comparação de que as habilidades são as microcompetências, ou como se as competências fossem as macro-habilidades. Nesta tese, sugerem-se as seguintes habilidades (mínimas) que o sujeito deverá apresentar para desenvolver o PC: compreensão, abstração, resolução de problemas, resolução algorítmica e avaliação, como se explica na sequência.

A primeira delas é a ‘compreensão’. Acredita-se que este é o primeiro passo para a resolução de um problema. Segundo Opmanis, Dagiene e Truu (2006, p. 5, tradução nossa), “um aspecto importante de qualquer tipo de manipulação de informações é a capacidade de compreender o significado da informação (em oposição ao processamento meramente mecânico)”. Inicialmente, o sujeito precisa entender (compreender) o problema, quer dizer, o que está sendo solicitado e qual o contexto. A partir daí, ser capaz de significar e aplicar, estabelecendo uma organização hierárquica entre compreensão e a habilidade de abstração. Como resultado da falta de compreensão, o indivíduo não consegue resolver o problema, pois a partir da leitura não é capaz de identificar quais são as partes importantes (abstração).

Na taxonomia de Bloom, em sua edição revisada, a palavra ‘compreensão’ foi substituída por ‘entender’, mantendo o mesmo significado que é de relacionar uma conexão entre o novo e o conhecimento previamente adquirido, em que a informação consegue ser reproduzida pelo aprendiz, pois foi entendida (KRATHWOHL, 2002). Portanto, a compreensão passa por uma aprendizagem significativa, na qual o sujeito constrói seu conhecimento e os seus processos cognitivos levam à resolução bem sucedida do problema. No primeiro passo, a partir da leitura, o sujeito precisa criar uma representação mental do problema, sendo o mais objetivo possível. Isso quer dizer: ler e compreender o que foi lido, podendo utilizar-se de seus conhecimentos prévios (subsunçores), pois, para Ausubel (2003, p. 7), é o “fator determinante do processo de aprendizagem”. Além disso, é necessário que o sujeito encontre a relação entre os conhecimentos prévios e os utilize para o passo seguinte, no qual deverá obter a solução do problema, e para isso necessitará de outras habilidades (apresentadas a seguir, como, por exemplo, abstração e resolução de problemas).

Corroborando com o entendimento sobre a habilidade de compreensão, Roazzi e Hodges (2013, p. 4) discorrem que “compreender torna-se, portanto, uma atividade de construção, interativa e ativa, que requer a integração de informações novas, contidas não apenas no texto, mas também no interior das estruturas de conhecimento do sujeito que está lendo”. Pode-se perceber nesta definição a presença dos elementos da teoria da aprendizagem significativa.

Em suma, a ‘compreensão’ é uma habilidade proposta nesta tese, a fim de apoiar os constructos definidos, considerada de grande importância andragógica, através da qual se pode perceber o entendimento do sujeito frente à resolução de um problema, que é um dos fatores críticos para o desenvolvimento do PC.

A segunda habilidade é a ‘abstração’, que no Currículo de Referência em Tecnologia e Computação (CRTC) foi definida como envolvendo a “[...] filtragem dos dados e sua classificação, ignorando elementos que não são necessários, visando aos que são relevantes. Envolve também formas de organizar informações em estruturas que possam auxiliar na resolução de problemas” (CRTC, 2018, p. 19). Em 2006, Wing (2006) definiu ‘abstração’ como sendo o processo para decidir quais os detalhes que deverão ser destacados e quais descartados. Onze anos depois, a mesma autora complementa que a abstração é o processo de mais alto nível quando se pensa em PC (WING, 2017). Neste ponto, há uma discordância, pois se acredita que para o sujeito executar o processo de abstração precisa necessariamente ter compreendido o problema, ficando, desta forma, a abstração como o próximo passo. A partir da abstração, pode-se determinar os pontos importantes e descartar os dados que não são relevantes, assim avançando para o próximo passo que é a resolução de problemas. A abstração é considerada importante e também difícil de ser desenvolvida, principalmente no tema que trata sobre resolução de problemas, confirmam França e Tedesco (2015).

A habilidade ‘resolução de problemas’ centra-se na capacidade do sujeito de encontrar uma ou várias soluções, independente do formato da sua resposta. Assim sendo, pode abarcar a decomposição de problema entendida como “o processo pelo qual os problemas são divididos em partes menores e mais fáceis de resolver” (CRTC, 2018, p. 19), quebrando em partes menores e gerenciáveis (ISTE; CSTA, 2011). Da mesma forma, e justificando sua importância, Ausubel, Novak e Hanesian (1983, p. 448, tradução nossa) descreveram que “a capacidade de resolver problemas constitui a meta primária da educação”.

Para entender a habilidade de ‘resolução algorítmica’, deve-se resgatar os conceitos apresentados na seção 3.3 sobre definir o que é um algoritmo e o que é um programa. Os algoritmos são uma sequência de passos lógicos, descritos em qualquer linguagem, geralmente em linguagem natural (por exemplo, língua portuguesa). Já os programas (ou *softwares*) são a “tradução” dos algoritmos para uma linguagem de programação. Portanto, essa habilidade está na prática do sujeito, a partir da habilidade anterior, na qual encontrou a solução do problema, traduzindo para um algoritmo ou *software*.

A ‘avaliação’ é a última habilidade proposta, a qual é entendida como a capacidade do sujeito após apresentar a resolução do problema através de algoritmo e/ou programa, de testar

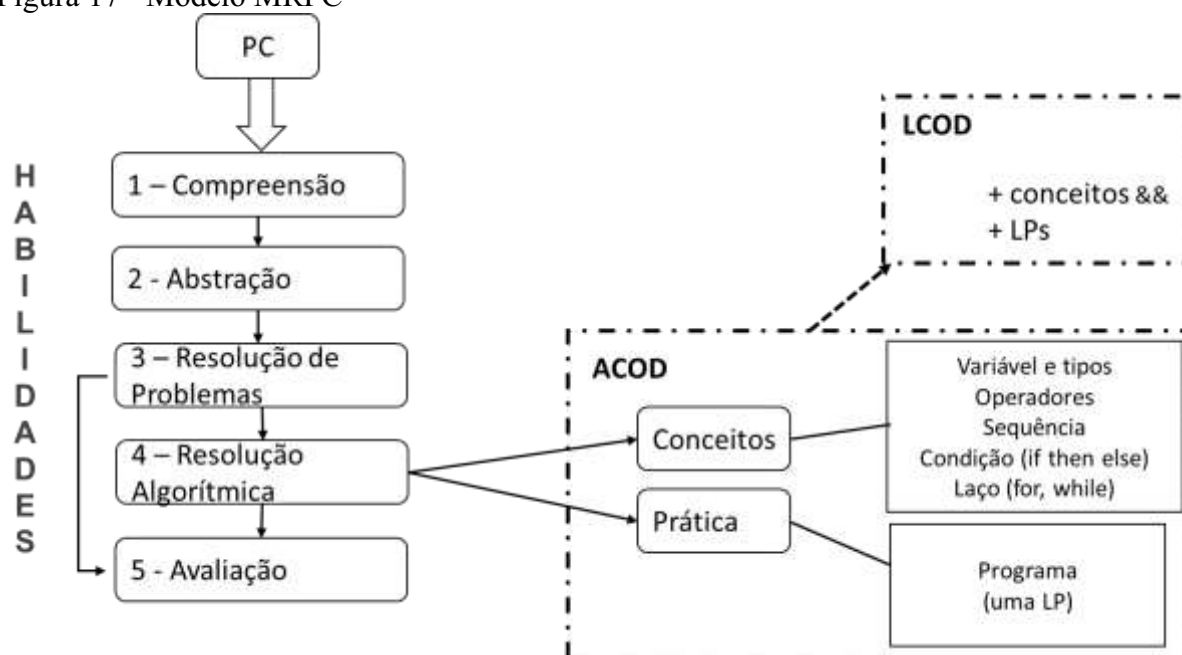
e avaliar a sua corretude. Em outras palavras, é entender o que foi implementado, ou seja, é a aptidão de analisar um código-fonte ou executável quanto às suas saídas. Também, entende-se como a avaliação ou teste de uma proposta de solução de um problema apresentado, independente do seu formato.

A partir de uma organização lógica das habilidades, propõe-se um modelo de referência, descrito detalhadamente na seção seguinte.

4.2.2 Modelo de referência para o desenvolvimento do PC – MRPC

Considera-se que as habilidades já citadas sejam as mínimas e necessárias a serem utilizadas para que o indivíduo seja capaz de resolver um problema e testá-lo. Na Figura 17, o modelo de referência para o desenvolvimento do PC (MRPC) está representado de forma gráfica.

Figura 17 - Modelo MRPC



Fonte: Da autora (2019).

Acompanhando a Figura 17, pode-se perceber a existência de uma dependência lógica, sugerida e não obrigatória, para desenvolver o PC. Em outras palavras, a partir de um problema, a primeira habilidade que deve ser mobilizada é a compreensão, pois sem que o sujeito entenda o problema não será capaz de resolvê-lo. Quanto ao que se entende como problema, este pode ser proposto por alguém ou percebido pelo próprio sujeito, pois esse tipo de pensamento

(computacional) não está somente restrito a resolver um problema, como também de definir uma situação que exige uma solução.

A partir da compreensão apropriada, a abstração passa a ser uma habilidade que naturalmente será mobilizada. Ou seja, o sujeito, compreendendo a situação, espera-se que identifique os pontos importantes e descarte os quais não têm utilidade. Após esta organização, o sujeito passa a focar seus esforços na busca de uma solução adequada. Neste ponto, não é importante saber quais métodos ou técnicas foram utilizadas pelo sujeito para encontrar o resultado, ou suas representações. Quando encontra o resultado (habilidade 3), e este possa ser avaliado ou testado (habilidade 5), entende-se que o processo de desenvolvimento do PC aconteceu, porém de uma forma desplugada.

Caso o caminho escolhido pelo sujeito seja o de transformar a sua solução em um algoritmo (habilidade 4), definido como sendo o “entendimento, aplicação, avaliação e produção” de um conjunto de instruções realizadas passo a passo para resolver uma tarefa (BROWN, 2015, texto digital, tradução nossa), logo passa-se a necessitar do domínio de determinados conceitos e práticas, ambos baseados no *framework* de Brennan e Resnick (2012).

Os conceitos os quais o programador deverá conhecer são estes:

- a) **variável:** posição de memória que pode ser identificada através de um nome e armazena um dado, em um determinado instante. Pode assumir vários tipos, dependendo da LP utilizada, e seu valor pode ser alterado durante a programação;
- b) **constante:** definida igualmente como uma variável, com a diferença que seu valor não pode ser alterado durante a execução do programa (DIAS, 2015);
- c) **operadores:**
 - c1) matemáticos: soma, diferença, divisão, multiplicação, potência, entre outros;
 - c2) lógicos: “e” (*and*) e “ou” (*or*);
 - c3) relacionais: maior, menor, maior igual, menor igual, igual, entre outros.
- d) **estrutura condicional:** representado pelos comandos IF/THEN/ELSE;
- e) **estruturas de repetição:** representado pelos comandos FOR e WHILE;
- f) **passos com sequência lógica baseados em eventos.**

Porém, não basta apenas conhecer esses conceitos de programação e, sim, também, saber aplicá-los (momento da prática). Como resultado, obtém-se um programa ou *software*. Em outras palavras, significa que, a partir da proposta de solução de forma algorítmica utilizando esses conceitos, é preciso que este algoritmo seja traduzido para uma linguagem de programação, tornando-se, então, um *software*.

A partir do programa, ou do algoritmo, deve-se fazer os testes e verificar se o resultado está correto; portanto, a habilidade de avaliação também é necessária, conforme pode ser visualizado no MRPC.

Ao apresentar a habilidade 4 e a partir de instrumentos avaliativos propostos e descritos na seção 4.2.4, infere-se para o sujeito o *status* de ACod, ou seja, está alfabetizado em código. Cabe salientar (que pode ser retomando na Figura 12) que o reconhecimento do sujeito neste estado exige como pré-requisito a AD (alfabetização digital). Porém, esse estado não pode ser atribuído com apenas poucas situações.

Em uma primeira situação, considerando que o sujeito/aprendiz nunca teve contato com a habilidade 4, e com programação, o processo de ensino e aprendizagem deve ser feito de forma a tornar significativos os conceitos e relacioná-los com a prática, principalmente ao considerar-se a aprendizagem significativa em adultos.

Quando o sujeito extrapola esses conhecimentos, ou seja, é capaz de implementar a sua prática em outras linguagens de programação, pode ser considerado como letrado em código (LCod).

A partir do entendimento desse modelo, levando-se como base a aprendizagem significativa, a andragogia e o contexto dos alunos de cursos binacionais, foi desenvolvida uma proposta de curso. Para esse curso, foram planejados materiais instrucionais, organizados em documentos (tutoriais, códigos-fonte, entre outros) para utilizar nas aulas e instrumentos para avaliação, todos apresentados no texto que segue.

4.2.3 Constructo das aulas

Foi proposto o curso “Eu Programo 1.0!”, com duração de 20 horas. A recomendação é que a cada aula um ou mais conceitos sejam apresentados ao aprendiz de forma rápida, quer dizer, não mais do que 10 minutos de apresentação, e segue-se a sua aplicação, em suma, a prática (programação). Knowles (1973) observou que as informações lembradas são as recebidas nos primeiros 15 minutos de aula. Já para Cavalcanti (1999, texto digital), os adultos “podem se concentrar numa explanação teórica durante 07 minutos. Depois disso, a atenção se dispersa”. Ainda, esse mesmo autor complementa que, neste intervalo de tempo, o professor poderá utilizar para apresentar “os objetivos e a relevância do assunto a ser discutido, enfatizar o valor deste conhecimento e dizer o quanto se sente motivado a discuti-lo” (texto digital).

Para executar a prática com os alunos, foi escolhido o *software App Inventor*⁵⁵. A escolha dessa ferramenta foi porque, a partir de uma interface simples, intuitiva, o usuário é capaz de desenvolver seus aplicativos em pouco tempo. Em outras palavras, já nos primeiros encontros o aluno consegue desenvolver um aplicativo simples, para o sistema operacional Android⁵⁶ (que possui a maioria dos usuários). Lembrando que o *App* é um *software* desenvolvido para dispositivos móveis; logo, a avaliação (habilidade 5) pode ser executada no próprio dispositivo do aluno.

O desenvolvimento de cada aula, basicamente, deve observar o seguinte roteiro: apresentação e/ou retomada dos conceitos; construção do App e teste, que podem acontecer de forma paralela. Após construir o App em aula, sempre é sugerido ao aluno um desafio (a fim de incentivar a aprendizagem por descoberta). Este consiste em propor uma melhoria ou alteração no aplicativo, a fim de verificar se houve aprendizagem por parte do discente. Quer dizer, parte-se do todo para as partes, conforme sugerem Knowles, Holton e Swanson (2011, p. 233): que “organização do conhecimento deve ser a preocupação essencial do professor ou de quem planeja a educação, de maneira que o trajeto do simples ao complexo não ocorra de partes arbitrárias e sem sentido a todos significativos, mas sim de todos simplificados a todos mais complexos”.

Os dois primeiros aplicativos não utilizaram nenhum conceito sugerido no MRPC, pois foram construídos para que o usuário entenda o funcionamento da ferramenta. Para os demais buscaram-se temas que fossem da realidade do aluno e dos seus interesses, pois, conforme Knowles, Holton e Swanson (2011, p. 74), “os adultos ficam prontos para aprender as coisas que têm de saber e para as quais precisam se tornar capazes de realizar a fim de enfrentar as situações da vida real”.

Em síntese, quanto à escolha dos temas dos aplicativos, o modelo andragógico de Knowles foi um dos principais referenciais, assim fomentando no aprendiz a necessidade do saber.

⁵⁵ Ferramenta de criação de *App*, disponibilizada gratuitamente na internet pelo MIT (disponível em: <http://ai2.appinventor.mit.edu/>). Nas palavras do prof. da Unicamp, Eduardo Valle, em 1º de maio de 2016, no jornal da Unicamp: "A facilidade da linguagem de programação visual trouxe essa potencialidade, de abrir a criação de aplicativos para um público não especialista, um público leigo". Sendo assim, estimula o interesse e a curiosidade que poderão levar a estudos mais avançados, continua o professor da Unicamp, que traduziu ferramenta do MIT para criação de aplicativos. Disponível em: https://www.unicamp.br/unicamp/sites/default/files/jornal/paginas/ju_653_paginacor_04_web.pdf. Acesso em: 12 set. 2018.

⁵⁶ Este sistema representou 74,45% do setor mundial de venda, entre 01/2018 e 01/2019, segundo a *Statcount*. Disponível em: <https://pcworld.com.br/ios-vs-android-quem-domina-o-mercado-de-smartphones/>. Acesso em: 15 ago. 2019.

Para cada App proposto, o professor, ao terminar seu encontro, disponibiliza uma possível solução para o problema. Quando o aluno termina o seu App, envia para o professor, que faz a correção e retorna um *feedback*, porque “sabemos que a andragogia deve ser mais personalizada para se adequar à singularidade entre os adultos” (KNOWLES; HOLTON; SWAMSON, 2011, p. 146), assim o foco da aprendizagem está no aluno e cabe ao professor/tutor acompanhar esse processo.

Todos os materiais construídos foram disponibilizados para os sujeitos em um ambiente virtual de aprendizagem (AVA), que serviu tanto de repositório de materiais quanto para a comunicação dos alunos com o professor. Esse ambiente se encontra no endereço eletrônico: <http://ecompartindo.santana.ifsul.edu.br/>, acesso com o usuário “banca” e senha “banca”, no curso “Eu Programa 1.0!”, acessando a edição “Eu Programa 1.0! – *Template* da Metodologia”.

Esse ambiente virtual permite também a implementação da Comunidade de Prática⁵⁷ (CoP). Na proposta da tese, esta comunidade pode ser operacionalizada com a ferramenta tecnológica do AVA, que é o Fórum, interagindo de forma assíncrona. A mesma interação também pode acontecer em uma ferramenta síncrona, a qual pode ser implementada através da criação de um grupo no *WhatsApp*®.

Considerando o posicionamento de Ausubel (2003, p. 21), quando descreve que “a aprendizagem por recepção verbal não é necessariamente memorizada ou passiva (tal como o é frequentemente na prática educacional corrente), desde que se utilizem métodos de ensino expositivos baseados na natureza, condições e considerações de desenvolvimento que caracterizam a aprendizagem por recepção significativa”, as aulas foram planejadas, juntamente com os materiais instrucionais, na sua maioria, para serem verbal-expositivas, ou seja, o professor, junto com os alunos, vai construindo o App, explicando e questionando cada comando ou grupo de comandos executados. Ainda, houve preocupação com a recomendação sobre o cuidado quanto aos seus materiais de instrução, além de atender a autonomia do aluno (KNOWLES; HOLTON; SWAMSON, 2011).

⁵⁷ Esta denominação e seus respectivos conceitos sobre Comunidade de Prática (CoP) foram propostos por Wenger (2002), a partir de seus estudos sobre modelos de aprendizagem. Atualmente é muito utilizado na área de administração de empresas, pois auxilia os indivíduos a desenvolver não somente a aprendizagem como também as competências organizacionais. Essa autora define CoP como sendo um grupo de pessoas que compartilham uma paixão sobre algo que sabem fazer e que interagem para aprender como fazê-lo melhor (WENGER, 2002). É gerar e compartilhar conhecimento. Para ser uma CoP precisa, obrigatoriamente, possuir: domínio (área de interesse do grupo compartilhando as questões principais), comunidade (membro que interagem neste grupo) e prática (compartilhamento de técnicas, métodos, experiências, formas de resolução de problemas etc.). Quanto à CoP ter seu papel como contribuição para a aprendizagem, os participantes, tanto professores como alunos, em uma posição de horizontalidade, ambos assumem que são educadores e educandos, sendo este conceito bastante pertinente na andragogia. Wenger (2002) enfatiza que as CoP objetivam a resolução de problemas, sendo que todos que participam aprendem, e essa participação precisa ser efetiva.

A partir das definições no MRPC, da escolha das ferramentas, do entendimento sobre os sujeitos e seus contextos, definiu-se o documento Plano de Aulas, o qual está disponível no AVA. Neste consta, entre outros dados, o número de horas do curso, seus objetivos e recursos necessários. Na sequência, para cada encontro foram definidos e detalhados os itens:

- a) **Objetivo**, como exemplos: aprender a usar as ferramentas tecnológicas (ferramenta *App Inventor* e AVA); conhecer e definir o conceito de variáveis;
- b) **Tema**, por exemplo: eventos simples, componentes visíveis e não visíveis, etc.;
- c) **Atividades práticas**, como, por exemplo: geração do App para salvar no dispositivo móvel; teste do aplicativo e;
- d) **Material** (ou recursos): um guia no qual toda a descrição do aplicativo é apresentada passo a passo, e também o App desenvolvido, em outras palavras, o código-fonte disponibilizado, com a extensão AIA.

No documento Guia de Aula, estão contidas todas as informações sobre o encontro, bem como a construção do App correspondente àquele encontro, com o respectivo código-fonte inteiramente comentado. A proposta é que, caso o aluno não tenha assistido à aula, seja capaz de acessar o material disponibilizado, desenvolver o seu aplicativo e entender os conceitos ali apresentados de forma autônoma.

O resumo do Plano de Aula com seus temas e práticas é mostrado no Quadro 7.

Quadro 7- Resumo dos Conceitos e Práticas dos App

Nome do Aplicativo	Tema/Conceito	Prática
Gato	Uso das ferramentas tecnológicas	-uso do ambiente da ferramenta: menu, <i>palette</i> , <i>viewer</i> , <i>Components</i> , <i>Properties</i> , abas <i>Projects</i> e <i>Connect</i> , Interface (<i>Designer</i>) e <i>Blocks</i> Editor (programação).
Tradutor	- eventos simples - seleção de componentes no <i>Designer</i> e comportamento no <i>Blocks</i> Editor; - componentes visível e não-visível; - manipulação de Propriedades dos componentes; - salvar <i>App</i> no dispositivo móvel.	- construção de um aplicativo (<i>App</i>) que utiliza os temas citados; - teste do <i>App</i> ; - geração do <i>App</i> para salvar no dispositivo móvel; - desafio para os alunos para aumentar as funcionalidades do <i>App</i> (repete para todos).
Combustível	- início da Programação: variável (global e local), comando de condição (<i>if/then/else</i>), operadores lógicos, aritméticos e relacionais.	- uso da aba <i>Projects</i> e <i>Connect</i> , Projeto de Interface (<i>Designer</i>) e <i>Blocks</i> (programação); - testes.

Nome do Aplicativo	Tema/Conceito	Prática
SMS – Dirigindo	- programação de variável, comando condicional, eventos do sensor de localização, concatenação de comandos.	- uso da aba <i>Projects</i> e <i>Connect</i> , <i>Designer</i> e <i>Blocks</i> ; uso da Paleta de Mídia (“TextoparaFalar”) e Social (“mensagensms”); - testes.
Pedômetro ou Contador de Passos e Calorias	- programação de variável, comando condicional, eventos do sensor de caminhada, concatenação de comandos, cálculo matemático.	- uso da aba <i>Projects</i> e <i>Connect</i> , <i>Designer</i> e <i>Blocks</i> ; uso da Paleta de Sensores (Pedômetro) e Mídia (som); - testes.
Calculadora	- programação de variável, comando condicional, comando de repetição, concatenação de comandos, cálculo matemático.	- uso da aba <i>Projects</i> e <i>Connect</i> , <i>Designer</i> e <i>Blocks</i> ; - testes.
Fatorial e Câmbio	- programação de variável, comando de repetição, operadores matemáticos e lógicos, cálculo matemático.	- buscar o <i>App</i> na Galeria do <i>App Inventor</i> (repositório) e compartilhar seu aplicativo com a comunidade. - uso da aba <i>Projects</i> e <i>Connect</i> , <i>Designer</i> e <i>Blocks</i> ; - testes.

Fonte: Da autora (2019).

Os dois primeiros aplicativos, Gato e Tradutor, foram planejados com o objetivo de que o aluno conheça a ferramenta de programação e já inicie o processo de pensamento de forma a contemplar as habilidades necessárias, isto é, o conceito de sequência lógica e sua prática.

O contexto do App Combustível refere-se ao tema sobre os carros do tipo Flex, no qual o aplicativo calcula para o usuário se é mais vantajoso colocar gasolina ou álcool no seu veículo, dependendo do preço praticado. O aplicativo seguinte permite ao motorista – quando estiver dirigindo, impossibilitado de atender a uma ligação no seu dispositivo móvel – enviar um SMS⁵⁸ ao número que está chamando, com uma mensagem previamente definida por ele. Estes aplicativos fazem parte de situações tipicamente de adultos.

O aplicativo ‘Contador de Passos’ pode ser utilizado por qualquer pessoa e foi projetado baseado no Programa *Active 10*⁵⁹, que sugere que diariamente uma pessoa caminhe 10 mil passos, a fim de manter sua saúde. Este aplicativo, além de contar os passos, calcula a distância percorrida e a quantidade de calorias consumidas.

⁵⁸ Do inglês *Short Message Service (SMS)*, que em português é traduzido para Serviço de Mensagens Curtas, muito utilizados em dispositivos móveis para comunicação.

⁵⁹ Mais informações disponíveis em: <https://www.bbc.com/portuguese/geral-42886546>. Acesso em: 10 ago. 2018.

A calculadora foi projetada para introduzir os conceitos sobre operadores matemáticos, além de trabalhar com os conceitos de condição. Por exemplo: mostrar ao usuário uma mensagem que a operação de divisão não poderá ser executada pelo número 0. Já o App Fatorial, foi planejado para calcular o fatorial de um número, porém utilizando diferentes comandos de repetição. Na primeira resolução o aluno deve implementar este cálculo utilizando o comando FOR, na segunda resolução o comando WHILE.

E, por fim, ‘Cambio’ é um aplicativo muito útil e totalmente coerente com a realidade dos sujeitos. Nas cidades-gêmeas, é comum em qualquer estabelecimento comercial fazer o pagamento em qualquer moeda, a saber, em Real ou Peso. Para começar a usar o aplicativo, deve-se informar o valor da cotação entre as duas moedas, conforme pode ser visualizado na Figura 18. Depois, informa o valor do produto (em pesos), o valor das notas que está pagando e em qual moeda, e o aplicativo calcula o valor do troco na mesma moeda indicada.

Figura 18 – Interface da App Cambio



Fonte: Da autora (2019).

A Figura 18 é a captura da tela em um dispositivo móvel, que representa o App tal como será mostrado para o usuário. No campo destinado a informar o valor da cotação, é executado somente uma vez, pois a cotação permanece enquanto o aplicativo estiver em uso.

Ao final do curso, está previsto o desenvolvimento de um programa (projeto final), no qual o aluno deverá executá-lo de forma individual. Porém, o tema deste não será descrito neste momento, pois dependerá de alguns fatores como o contexto dos alunos e seus interesses. Em

síntese, o tema do projeto final deverá ser combinado entre o professor e o grupo, pois há que se levar em consideração as experiências dos alunos e seus interesses tanto pessoais, quanto profissionais, uma vez que, quando apresenta a proposta para o aluno/adulto desenvolver o seu projeto, fica clara a finalidade nas palavras de Rocha (2012, p. 1): “criar espaço para que ele seja criativo e tenha iniciativa em suas ações de aprendizagem. O adulto aprende mais e melhor quando percebe que lhe é dada a autonomia para o seu crescimento pessoal e profissional”.

Caberá ao professor, a partir do levantamento realizado, dimensionar o projeto do *software*, para que este não extrapole a complexidade até então desenvolvida no curso. Isto significa que cabe ao professor analisar os requisitos, bem como os conceitos necessários para desenvolver o projeto final. Já que são alunos, adultos e autônomos, poderá também propor desafios neste projeto a serem pesquisados pelos alunos (aprendizagem por descoberta) não contemplados durante o curso, considerando Ausubel (2003, p. 22), quando afirma que “por conseguinte, deve ficar claro que a aprendizagem significativa constitui o meio principal para se aumentar o armazenamento de conhecimentos do aprendiz, quer dentro, quer fora da sala de aula”. Nesta proposta, entende-se que o professor é o facilitador da aprendizagem, pois ninguém ensina o outro a pensar, este ato é ímpar e pessoal.

Logo no início do curso, é sugerido que o professor apresente o contrato de aprendizagem (CAPre) para os alunos. Este documento serve para organizar e orientar sobre os objetivos dos processos de ensino e aprendizagem para todos os envolvidos, fomentando a autonomia do aprendiz⁶⁰ (KNOWLES; HOLTON; SWAMSON, 2011). O modelo definido nesta tese segue as sugestões dos autores anteriores e está disponibilizado no AVA, bem como está de acordo com o curso definido e seus objetivos. Cabe salientar que o professor apresenta este contrato aos aprendentes, que podem (re)construir o documento de forma colaborativa, conforme as necessidades e interesses do grupo.

Após o término do curso, para os alunos que participaram, como forma de reconhecer seu aprendizado, um certificado pode ser emitido, sendo que o *template* está disponível no AVA.

Somando-se à construção desses documentos e aplicativos apresentados, a fim de completar o curso, instrumentos avaliativos foram desenvolvidos e são discutidos na seção seguinte.

4.2.4 Instrumentos de avaliação

⁶⁰ Apresentado em detalhes na seção 2.3.2.1.

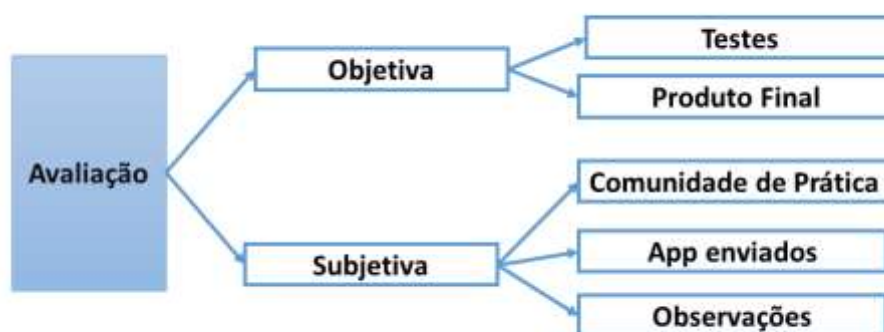
Para iniciar a abordagem sobre a avaliação, retoma-se o objetivo deste trabalho que é de compreender e analisar o desenvolvimento do PC nos sujeitos (que já completaram o pós-médio), entendido neste contexto como adultos. Para tanto, faz-se necessário acompanhar e avaliar este processo.

Como contribuição desta tese, são propostos alguns instrumentos avaliativos com a finalidade de responder a essa questão. Segundo Gil (2002), quando a pesquisa acontece por meio de estudo de caso, deve-se utilizar mais de uma técnica para coletar os dados. Em suma, o uso de procedimentos diversos para obter os dados é imprescindível para aumentar a qualidade dos resultados encontrados, o que também é corroborado por Yin (2001).

Os pesquisadores Brennan e Resnick (2012) propuseram sugestões para avaliar o desenvolvimento do PC através da programação. Embora seu trabalho esteja focado no nível K-12, julgou-se importante considerar alguns aspectos deles para esta tese. Por exemplo, a recomendação de que muitos projetos sejam desenvolvidos ao longo do processo a fim de oportunizar ao professor o entendimento do pensamento de cada aluno. Outra observação relevante considerada foi a existência de vários pontos de verificação da aprendizagem durante os encontros.

Como avaliar não é um processo simples, propõem-se vários instrumentos, os quais estão agrupados em dois conjuntos: instrumentos objetivos e subjetivos, conforme podem ser visualizados de forma gráfica na Figura 19.

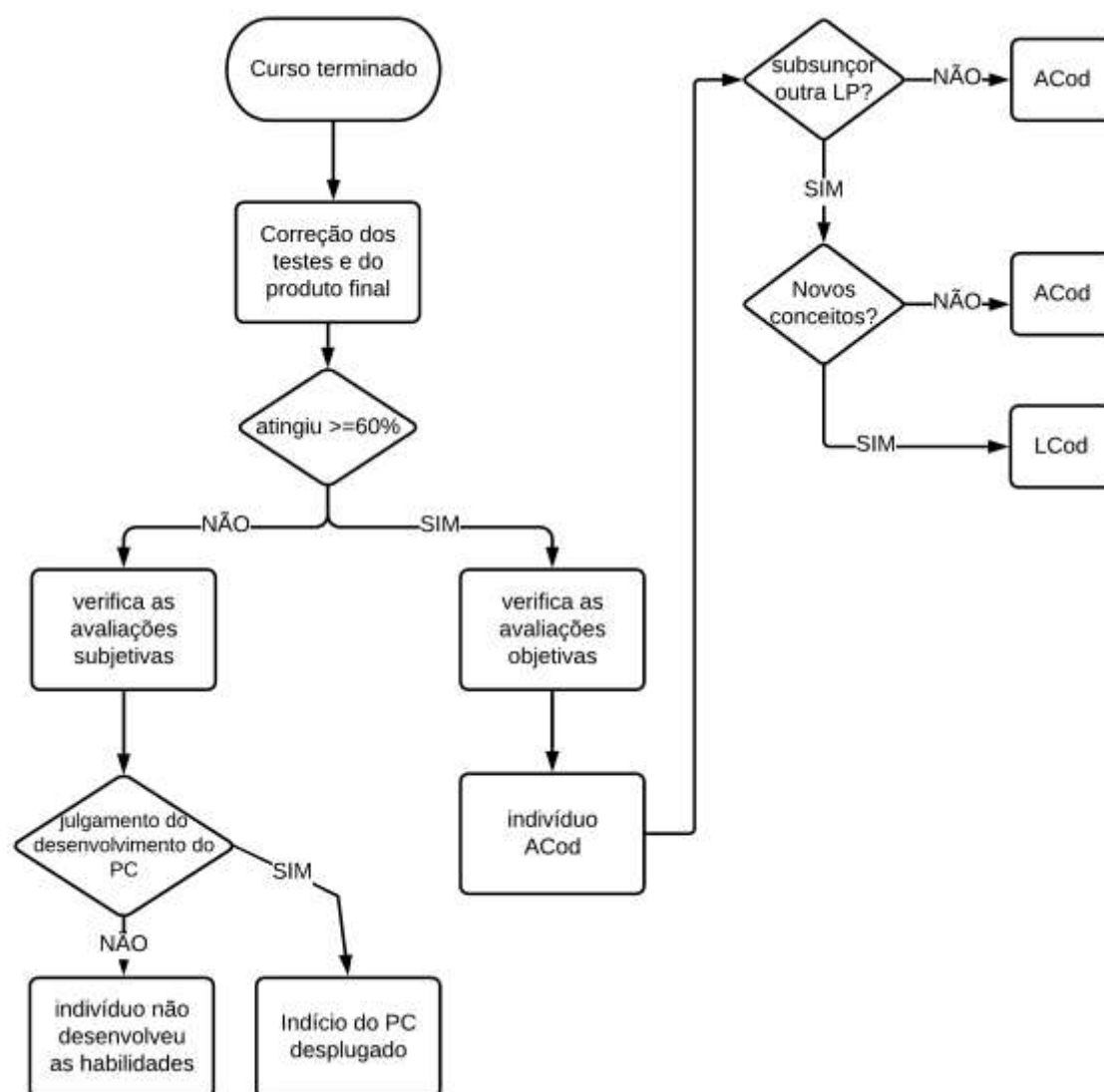
Figura 19 - Tipos de instrumentos avaliativos



Fonte: Da autora (2019).

Os (instrumentos) objetivos estão compostos por testes (pré, pós e intermediários) e pelo produto resultante do projeto final, com seus respectivos indicadores. Os instrumentos subjetivos são: observação, comunidade de prática (CoP) e aplicativos desenvolvidos durante o curso (estes últimos descritos no Quadro 7). Essa divisão serve para o entendimento da proposta avaliativa, de forma estática. Já na Figura 20, o processo de avaliação, através de um fluxograma, é mostrado.

Figura 20 - Processo de avaliação para o MRPC



Fonte: Da autora (2020).

Após todos os testes realizados e o projeto final (App) entregue e devidamente avaliado, fez-se a média dos valores encontrados, sendo que o resultado deverá atingir o valor mínimo de

60% de acertos. Esse valor de referência foi baseado na Organização Didática do IFSUL⁶¹ e também deve ser observado para os indicadores do produto final. Com esses valores conhecidos, pode-se sugerir o *status* do indivíduo quanto à alfabetização em código. Já para a avaliação subjetiva, o professor deverá avaliar de forma individual cada aluno e então poderá determinar se o discente desenvolveu todas habilidades propostas no MRPC, juntamente com as observações realizadas durante todo o curso e na CoP. Então, o tutor poderá sugerir que o aluno desenvolveu o PC, porém de forma desplugada⁶², ou seja, a Habilidade 4 não foi desenvolvida. A seguir, será mostrado o detalhamento das avaliações.

4.2.4.1 Avaliação objetiva

São propostos o pré-teste (ou teste 1), um teste intermediário e o último chamado de pós-teste. As questões que compõem os dois primeiros testes foram retiradas do *Bebras* e, portanto, já validadas. Para cada questão já está anotado a qual das habilidades ela está atrelada. Por exemplo, observe a Figura 21.

Figura 21 - Exemplo da questão 2 do pré-teste

<p>A <i>International Society for Technology in Education</i> (ISTE) enfatiza que o pensamento computacional na educação não está trazendo alunos para a posição de líder em Ciência da Computação, mas aplicando suas habilidades de pensamento computacional também em outros cursos como um hábito, aumentando a capacidade do indivíduo em resolver problemas do seu cotidiano.</p>
<p>Por isso, vamos resolver as questões!! Essas questões foram retiradas do teste Bebras. Estão disponíveis para responder no AVA (http://ecompartindo.santana.ifsul.edu.br/) ou envie sua resposta no WhatsApp (privado!! Com seu nome e as respostas).</p> <p>A cor azul significa que o texto está escrito em espanhol e em verde na língua portuguesa. Se você não entendeu a questão e não sabe como resolvê-la, marque a alternativa NÃO ENTENDI A QUESTÃO.</p>
<p>2. Garrafas/ Botella</p>
<p>Un castor pone cinco botellas en una mesa. Las coloca para que cada botella se vea un poco. Pone la primera en la parte de atrás de la mesa y coloca cada botella subsecuente delante de la anterior.</p> <p>Um castor coloca cinco garrafas em uma mesa. Ele as coloca de modo que cada garrafa apareça um pedaço. Ele coloca a primeira garrafa na parte de trás da mesa e cada garrafa subsecuente na frente da anterior.</p>

⁶¹ Organização Didática da Educação Básica, Profissional e Superior de Graduação. Disponível em: <http://www.ifsul.edu.br/regulamento-da-atividade-docente/item/113-organizacao-didatica>. Acesso em: 10 jan. 2020.

⁶² Neste contexto, o PC desplugado significa que o aluno não desenvolveu a Habilidade 4, ou seja, não resolveu os problemas de forma algorítmica ou através de programas de computador.



¿En qué orden fueron colocadas las botellas según la imagen?

Em que ordem as garrafas foram colocadas conforme mostrado na figura?

- a) E D C B A
- b) D B C A E
- c) E C D A B
- d) D C E B A
- e) não entendi a questão / no entendí la pregunta

País proponente: Inglaterra

Habilidade: resolução de problema, abstração

Fonte: Da autora (2019).

Na figura anterior, pode-se perceber que as questões são apresentadas em dois idiomas, o país que desenvolveu a questão e quais as habilidades envolvidas. Neste teste, bem como nos demais, foi acrescentada a alternativa “não entendi a questão”, porque tem-se a Habilidade 1 (compreensão) na avaliação. Nos testes, as alternativas são de múltipla escolha, por isso são considerados como instrumento objetivo. Na Figura 22, a primeira questão do segundo teste é mostrada.

Figura 22 - Questão 01 – Teste 2

T02.01: Relógio Quebrado/ Reloj Roto

Um relógio digital exibe quatro dígitos. Cada dígito é mostrado usando sete segmentos que podem estar ligados ou desligados, conforme a figura abaixo.

Un reloj digital muestra cuatro dígitos. Cada dígito se muestra usando siete segmentos que pueden estar prendidos o apagados, según la figura siguiente:



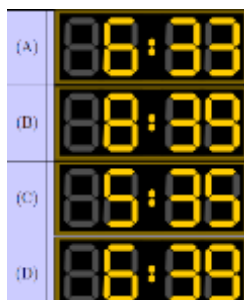
O relógio quebrou exatamente em um dos sete segmentos de um dígito.

El reloj se rompió exactamente en uno de los siete segmentos de un dígito.



Se o relógio quebrou e estava exibindo a hora acima, qual das seguintes opções pode ser a hora real?

Si el reloj se rompió y presentaba la hora 6:35. Cual de las siguientes opciones es correcta para la hora?



E) Não entendi a pergunta / No entendí la pregunta

País proponente: Japão - Habilidade: resolução de problemas, avaliação, abstração

Fonte: Da autora (2019).

Na Figura 22, pode-se observar um outro exemplo proposto para os alunos. Para a resolução deste teste, três habilidades são mobilizadas: a resolução de problemas (Habilidade 3), avaliação (Habilidade 5) e abstração (Habilidade 2), conforme o MRPC. Os demais testes estão disponibilizados no ambiente virtual.

Já o pós-teste diferencia-se por ser mais extenso, pois é composto de cinco questões. Inicialmente é apresentado um texto com explicações sobre a importância do índice de massa corporal para a saúde do corpo. Na Figura 23, está evidenciada a descrição do problema do pós-teste (primeira lauda).

Figura 23 - Início do pós-teste

Descrição: IMC é a sigla para **Índice de Massa Corporal** que serve para avaliar o peso do indivíduo em relação à sua altura e assim indicar se está dentro do peso ideal, acima ou abaixo do peso desejado. Estar dentro do peso certo é importante porque estar acima ou abaixo do peso influencia na saúde, aumentando o risco de doenças como desnutrição quando se está abaixo do peso, e AVC e infarto, quando se está acima do peso. O cálculo do IMC deve ser feito usando a seguinte fórmula matemática:

Índice de Massa Corporal

$$\text{IMC} = \frac{\text{Peso (Kg)}}{\text{Altura (m)}^2}$$

O que significam os resultados do IMC

Cada resultado do IMC deve ser avaliado por um profissional de saúde. A tabela a seguir indica os possíveis resultados do IMC, de acordo com a Organização Mundial da Saúde, sendo que o IMC entre 18,5 a 24,9 representa o peso ideal e o menor risco de algumas doenças. **Tabela 01 - Resultado**

Classificação	IMC	O que pode acontecer
Muito abaixo do peso	16 a 16,9 kg/m ²	Queda de cabelo, infertilidade, ausência menstrual
Abaixo do peso	17 a 18,4 kg/m ²	Fadiga, stress, ansiedade
Peso normal	18,5 a 24,9 kg/m²	Menor risco de doenças cardíacas e vasculares
Acima do peso	25 a 29,9 kg/m ²	Fadiga, má circulação, varizes
Obesidade Grau I	30 a 34,9 kg/m ²	Diabetes, angina, infarto, aterosclerose
Obesidade Grau II	35 a 40 kg/m ²	Apneia do sono, falta de ar
Obesidade Grau III	maior que 40 kg/m ²	Refluxo, dificuldade para se mover, escaras, diabetes, infarto, AVC

Interface (Tela):

Sexo
 Mulher Homem
 help
 Idade
 anos
 help
 Altura
 m
 help
 Peso
 kg
 help
 Atividade Física
 Sedentária Moderada Intensa
 help

Fonte: Da autora (2019).

Na Figura anterior, os dados e informações da fórmula de cálculo e a interface do aplicativo esperada para resolver o problema estão descritas, bem como a classificação do IMC. A partir dessa contextualização do problema, quatro questões são especificadas, sempre considerando as habilidades do MRPC, na Figura 24, que mostra a continuação do teste, na versão escrita em língua portuguesa.

Figura 24 - Questões do pós-teste

Você foi convidado a desenvolver este APP para calcular o IMC e mostrar o valor do IMC resultante e sua classificação. Baseado na explicação anterior, responda as questões 01, 02, 03 e 04. Marque somente uma alternativa.

- 1. A interface (tela) está totalmente de acordo com a descrição do problema? (1,2)**
 - a) não entendi a questão
 - b) sim
 - c) não
 - d) em partes
- 2. A tabela 01 apresenta dados que são importantes para a programação do APP. (1,2,3)**
 - a) não entendi a questão
 - b) sim
 - c) não
 - d) em partes
- 3. Na interface proposta, é possível remover os botões de Sexo (Mulher e Homem) sem alterar a funcionalidade do aplicativo? (1,2,3,4,5)**
 - a) não entendi a questão
 - b) sim
 - c) não
 - d) em partes
- 4. A figura abaixo mostra a fórmula do IMC implementada e parte da implementação da Tabela para mostra o resultado ao usuário. Esta porção do código está totalmente correta? 1, 2, 4**



- a) não entendi a questão
 - b) sim
 - c) não
 - d) em partes - Se contiver erro, descreva-o.
- 5. Qual o valor que será mostrado na interface no “CampoDeTextoMostrarResultado” (variável valor), segundo a figura abaixo?**



Resposta: _____

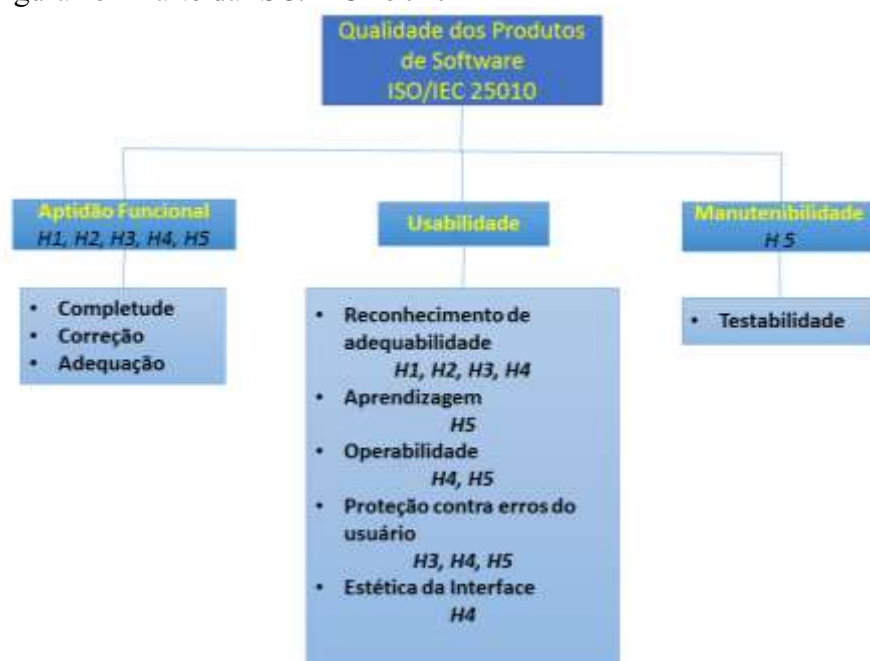
Pode-se perceber ao analisar a figura correspondente às perguntas do pós-teste que no final de cada questão aparecem entre parênteses alguns números que indicam a(s) habilidade(s) que estão sendo buscadas. Na pergunta 1, as habilidades de compreensão e abstração são analisadas; na segunda questão, as habilidades de compreensão, abstração e resolução de problemas; já na pergunta 3, todas as habilidades propostas no MRPC estão sendo examinadas; na questão seguinte a compreensão, abstração e resolução algorítmica estão sendo julgadas. Na última questão, busca-se verificar a habilidade de avaliação aliada ao conceito de variável e constantes.

O método de avaliação do projeto final (produto) está desenvolvido a seguir.

4.2.4.2 Avaliação do produto

Para avaliar o resultado do *App* proposto no projeto final do curso, buscou-se a norma ISO/IEC 25010 (2011)⁶³ (*System and Software Quality Models*), que apresenta a definição de oito características de qualidade de *software* e as suas subcaracterísticas que são apresentadas para determinar o escopo de cada medida. Na Figura 25, somente está apresentado um subconjunto desta, o qual é relevante para este trabalho.

Figura 25 - Parte da ISO/IEC 25010



Fonte: Adaptado pela autora com base na Norma ISO/IEC 25010 (2011).

⁶³ Disponível em: <https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en>. Acesso em: 2 ago. 2018.

Conforme pode ser visualizado na Figura 25, consideram-se três características da norma, que são Aptidão Funcional, Usabilidade e Manutenibilidade⁶⁴. Por Aptidão Funcional (*Functional Suitability*) entende-se como o grau em que um produto ou sistema fornece funções que atendam às necessidades declaradas e implícitas quando usadas sob condições especificadas. No contexto da tese, somente três subcaracterísticas são consideradas, e todas observam as cinco habilidades propostas no MRPC. São descritas como:

- **Completeness Funcional** (*Functional Completeness*): grau para o qual o conjunto de funções abrange todas as tarefas especificadas e objetivos do usuário;
- **Correção Funcional** (*Functional Correctness*): grau para o qual um produto ou sistema fornece os resultados corretos com o grau de precisão necessário e;
- **Adequação Funcional** (*Functional Appropriateness*): grau para o qual as funções facilitam a realização de tarefas e objetivos específicos.

Essas subcaracterísticas têm o foco na funcionalidade e, portanto, são obrigatórias de serem cumpridas (implementadas) em qualquer produto de *software*.

A característica Usabilidade (*Usability*) é definida como sendo o grau para o qual um produto ou sistema pode ser usado por usuários determinados para atingir metas especificadas com eficácia, eficiência e satisfação em um contexto específico de uso. É composto por seis subcaracterísticas, das quais cinco são consideradas, a saber:

- **Adequabilidade** (*Appropriateness Recognizability*) que é o grau em que os usuários podem reconhecer se um produto ou sistema é adequado às suas necessidades;
- **Aprendizagem** (*Learnability*) é o grau em que um produto ou sistema pode ser usado por usuários específicos para atingir as metas definidas de aprender a usar o produto ou sistema com eficácia, eficiência, isenção de riscos e satisfação em um contexto único de uso;
- **Operabilidade** (*Operability*) é o grau para o qual um produto ou sistema possui atributos que facilitam a operação e o controle;
- **Proteção contra erros do usuário** (*User Error Protection*) é o grau para o qual um sistema protege os usuários contra erros; e

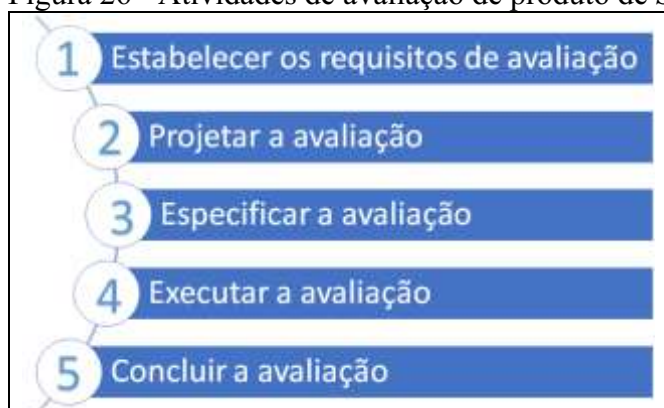
⁶⁴ Esta norma ISO/IEC não tem correspondente na ABNT (Associação Brasileira de Normas Técnicas); portanto, não está escrita em língua portuguesa. Todas as traduções foram feitas pela autora.

- **Estética da Interface do Usuário** (*User Interface Aesthetics*) entendida como o grau para o qual uma interface de usuário permite uma interação agradável e satisfatória para o usuário.

Por fim, a última característica abordada é a Manutenção (*Maintainability*), que representa o grau de eficácia e eficiência com o qual um produto ou sistema pode ser modificado para melhorá-lo, corrigi-lo ou adaptá-lo às mudanças no ambiente e nos requisitos. A subcaracterística correspondente considerada é a Testabilidade (*Testability*), definida como sendo a realização dos testes para determinar se os critérios especificados foram atendidos.

A partir das definições de cada uma das subcaracterísticas, bem como dos critérios de avaliação (Figura 25), o próximo passo é executar o processo de avaliação do produto de *software*. Este está descrito na ISO/IEC 25040, que detalha cinco atividades (Figura 26).

Figura 26 - Atividades de avaliação de produto de *Software*




Fonte: Adaptado e traduzido pela autora com base na norma ISO/IEC 25040.

Na primeira atividade, os critérios de qualidade, já apresentados na Figura 25, foram cuidadosamente selecionados. Na atividade 2, as questões (indicador) da presença ou não da subcaracterística avaliada. Já na atividade 3, um *template* para avaliação que poderá ser utilizado pelo professor é apresentado (Figura 27), bem como a escala da presença. O tutor ou avaliador do App final preenche o formulário, que está disponibilizado no AVA, marcando a presença ou ausência do indicador, sempre lembrando que na característica Funcionalidade é obrigatória a marcação positiva. Para concluir a avaliação, o professor deverá contar a presença dos indicadores em um total de 10. Pelo menos 6 deverão estar presentes (60%), sendo obrigatórios todos os da característica Funcionalidade. A partir desse resultado, o passo 5 será de emitir um parecer ou uma nota, a fim de concluir a avaliação.

Terminado esse processo, ambas as avaliações objetivas deverão ter atingido 60% para que o aluno seja considerado ACod.

A seguir, apresentam-se os indicadores e o modo de avaliar de forma subjetiva.

Figura 27 - Avaliação do Projeto Final



Avaliação do Projeto Final do Curso

Nome do APP:
 Nome do participante:
 Nome do Avaliador:
 Data:

Característica	Subcaracterística	Pergunta	Presença	
			Sim	Não
Aptidão Funcional	Completeness funcional	Propõe-se a fazer o que é apropriado? Tudo o que foi planejado?		
	Correção funcional	Faz o que foi proposto de forma correta?		
	Adequação funcional	Propõe-se a fazer o que é apropriado?		
Usabilidade	Reconhecimento de adequabilidade	Atendeu suas necessidades?		
	Aprendizagem	É fácil de aprender?		
	Operabilidade	É fácil de operar e controlar?		
	Proteção contra erros do usuário	Tem validação de dados de entrada?		
	Estética da interface do usuário	A interface é agradável?		
Manutenibilidade	Testabilidade	É fácil de testar?		
Total				

Fonte: Adaptado pela autora com base na ISO/IEC 25040.

4.2.4.3 Avaliação Subjetiva

São propostas três formas de avaliar o processo de desenvolvimento do PC, que são: observação direta, o acompanhamento dos aplicativos desenvolvidos durante o curso e a interação do aluno na CodP.

A observação direta é uma evidência que deve ser realizada pelo professor durante o transcorrer de suas aulas, e isso significa observar seus alunos em tempo real e tratando de refletir e julgar o posicionamento do estudante de acordo com o contexto e o conteúdo sendo trabalhado. Por conseguinte, sabe-se que é uma atividade que consome bastante tempo (YIN, 2001), sendo que o resultado deve ser o menos tendencioso possível. Sugere-se que o professor faça um diário de bordo com suas anotações.

No decorrer das aulas, o aluno deve ser observado, e quando um aplicativo for finalizado, ou mesmo no final de cada encontro, o professor solicita que o respectivo App seja disponibilizado tanto para ele quanto para o grupo de colegas. A partir do código-fonte, o professor poderá acompanhar a produção do aluno com relação ao *software* desenvolvido, verificar a completude, se houve erros, entre outros. Por conseguinte, fornecer um *feedback* adequado, pois, indubitavelmente, ao analisar o código-fonte, pode-se perceber a presença (ou não) das habilidades referentes àquele App. Acredita-se que, desse modo, o professor se torna capaz de traçar o perfil de conhecimento do seu aluno de forma individualizada.

Ainda, no viés da subjetividade, agregando a observação e a análise de desenvolvimento de cada aluno, cabe ao professor incentivar o uso da CoP, a fim de que o grupo compartilhe suas experiências (como já visto no acompanhamento dos App), suas dúvidas e seus acertos. A partir das interações dos alunos, cabe ao professor a observação dessas comunicações para compor o perfil do aluno com relação ao desenvolvimento do seu PC, através das habilidades sugeridas no MRPC. Por exemplo, se um aluno apresenta na CoP uma dúvida e algum colega responde, então o professor já terá dados sobre as habilidades dos dois membros interagentes.

A partir desses instrumentos, nos quais a atuação do professor é importante e fundamental, caberá a ele o julgamento sobre o conhecimento e o desenvolvimento do PC do aluno. O parecer versará sobre se há indícios de melhorias no PC do aluno ou se não foram possíveis evidenciar as habilidades sugeridas no modelo.

Em resumo, para todos os App propostos foram considerados, primeiramente, as experiências, aprendizagens anteriores e o contexto do aluno, a fim de ancorar os novos conhecimentos. A cada final de App, um desafio para melhoria é sugerido. Para reafirmar as escolhas dos temas propostos, Yin (2001) sugere que os estudos de casos estejam relacionados a algum contexto da vida real, o que vem ao encontro desta pesquisa quando o sujeito é um indivíduo adulto e seus interesses estão na sua vida cotidiana, seguindo as propostas andragógicas de Knowles (1981).

Ao considerar o sujeito como centro da aprendizagem, de acordo com Li (2014), argumenta-se que o PC deve estar incorporado em todo o processo de ensinar, nos quais os professores devem propor casos para serem resolvidos baseados nas ideias de PC. Além disso, afirma que os alunos podem gradualmente entender o pensamento abstrato e o raciocínio lógico através de treinamento do pensamento e aplicação de métodos de PC na prática, pois o conhecimento do indivíduo está na habilidade de resolver problemas reais. Com isso, justifica-se a quantidade de atividades práticas sugeridas nesta pesquisa.

De um lado, ancorado em Ausubel (2003, 1983) e Knowles, Holton e Swanson (2011) e, de outro lado, no entendimento e avaliação de PC, propôs-se estes constructos andragógicos, que tratam tanto de desenvolvimento de habilidades para resolver problemas na forma de programas ou não, quanto para avaliar este processo. O resultado deverá mostrar indícios das habilidades propostas, com a finalidade de afirmar que o sujeito tornou-se um ACod, ou desenvolveu as habilidades exceto a de programação (PCD – pensador computacional desplugado).

Em síntese, neste capítulo foram delimitados os caminhos metodológicos percorridos, bem como os instrumentos resultantes da pesquisa, que são: o MRPC, o método de avaliação para o modelo, um constructo andragógico para a aplicação, acompanhamento e avaliação dos instrumentos anteriores. Retomando, nos documentos deste constructo estão contidos o Plano de Aulas para os dois estudos de casos realizados, o contrato de aprendizagem, o questionário Conhecendo Você, todos os testes realizados (nos dois idiomas), os materiais instrucionais (Guias de Aula) e a implementação da CoP, todos disponibilizados no ambiente virtual de aprendizagem (AVA). A plataforma de aprendizagem utilizada para a implementação dos estudos de casos foi o Moodle (*Modular Object Oriented Dynamic Learning*), que é um *software* de código aberto.

No Capítulo seguinte será detalhado o Caso Piloto ou Estudo de Caso 1 (EC1) realizado.

5 ESTUDO DE CASO 1 (EC1)

Neste capítulo, está descrita a aplicação do constructo andragógico desta tese, ou seja, o caso piloto (YIN, 2001), complementando a Figura 15, destacando-se os passos 4 e 5. O caso piloto também será denominado, no decorrer do texto, de Estudo de Caso 1 (EC1).

A primeira experiência realizada foi com o Curso de *Tecnico Terciario en Logistica*, na *Universidad del Trabajo del Uruguay* (UTU), na cidade de Rivera. Este curso é binacional e tem duração de quatro semestres. Na matriz curricular, estão previstas duas disciplinas que abordam conteúdos relacionados à Informática (conforme a nomenclatura utilizada pela Instituição). No segundo semestre do curso, está prevista a oferta da disciplina Informática Aplicada I e, no semestre seguinte, a Informática Aplicada II, ambas com carga horária de 64 horas.

Inicialmente, para que o caso piloto fosse autorizado, foi entregue uma carta ao Diretor da UTU em Rivera, professor Ricardo Olivera, solicitando a sua permissão, cuja resposta foi positiva para a realização do estudo.

As intervenções aconteceram no período de setembro a novembro de 2018, no horário da disciplina Informática Aplicada I, de responsabilidade do professor Robert Gomez. Esta contava com 10 alunos matriculados, com média de idade de 23 anos, sendo que 40% são homens, 60% mulheres, e 80% do total são de nacionalidade uruguaia. Na ementa está especificado o conteúdo de Planilhas Eletrônicas. O professor responsável desenvolveu este tema anterior ao início do EC1.

Observando a disposição dos horários reservados da disciplina, foram planejados 13 encontros⁶⁵ (três horas de aula por semana). Porém, houve paralisações e greves, sendo que no final aconteceram somente 11 aulas, que corresponde a 16 horas e trinta minutos.

No primeiro encontro, foi explicado o objetivo do curso, sua importância no contexto do mercado de trabalho, o que é PC, o que são aplicativos e o conteúdo do TCLE. Também foi apresentada a plataforma do AVA com o cadastramento dos alunos e como utilizá-la.

Ainda no primeiro encontro, os alunos se mostraram receptivos e empolgados com a proposta do curso “Eu Programa 1.0!”. Além disso, foi apresentado o CAPre e todos concordaram com o seu conteúdo. O logotipo do curso foi desenvolvido para criar uma identidade visual do grupo com o curso e está mostrado na Figura 28.

Foi construído um grupo no aplicativo *WhatsApp* (com o mesmo logotipo) e no AVA foram disponibilizados fóruns que objetivaram a interação entre os alunos e com o professor, a fim de incentivar o trabalho colaborativo. Esses dois mecanismos de comunicação têm como objetivo a implementação da Comunidade de Prática (CoP) do tema, que é o foco do curso proposto (conforme descrito no Capítulo 4).

Figura 28 - Logotipo do Curso



Fonte: Da autora (2018).

Ficou disponível no AVA o questionário “Conhecendo Você”. O objetivo foi levantar informações sobre os conhecimentos sobre programação dos alunos e demais dados, a ser respondido quando os alunos considerassem mais oportuno.

As aulas aconteceram no Laboratório de Informática da UTU e algumas dificuldades quanto ao uso da tecnologia foram enfrentadas. Um dos problemas foi solucionado quando 5 *tablets* para o desenvolvimento dos App e dois aparelhos celulares para o teste foram disponibilizados. Em alguns momentos, foi necessário disponibilizar os dados móveis da pesquisadora para conseguir conexão com a internet. Esses detalhes, em determinados momentos, dificultaram o trabalho dos alunos, porque desviam a atenção e o foco do objetivo principal: desenvolver o PC e o App.

⁶⁵ No contexto deste trabalho as palavras encontro e aula possuem o mesmo significado.

Em resumo, a primeira aula foi basicamente para organizar o curso, fazer cadastramento no AVA, apresentações, verificação se todos já possuem *e-mail* válido, entre outros.

As aulas seguintes continuaram um padrão no seu início. Ao começar o encontro, fazia-se uma retomada da aula anterior de no máximo 10 minutos, pois, segundo orientações de Knowles (1973), 15 minutos são suficientes para reter a atenção do aprendiz. Após esse momento, a prática de desenvolvimento dos App ocorria. O formato de trabalho geralmente acontecia com os alunos dispostos em uma mesa compartilhada, conforme pode ser visualizado na Figura 29.

Figura 29 – Uma aula do EC1



Fonte: Da autora (2018).

No segundo encontro iniciou-se a programação do App do Gato, que foi somente finalizado no encontro seguinte. Percebeu-se que 4 alunos tiveram dificuldade em executar tarefas de encontrar e copiar arquivos (*download*) e enviar arquivos (*upload*). Para esses alunos foi dispensado um atendimento especial, visto que apresentaram dificuldade no que tange à alfabetização digital, pontual, no uso do AVA. Após esta intervenção, todos os alunos puderam ser considerados alfabetizados digitalmente.

Na sequência, foi implementado o App Tradutor, quando já se iniciaram os conceitos de variáveis e comando de condição. Conforme o planejamento, o próximo App a ser implementado seria o App SMS. Como já estávamos no quinto encontro, o planejamento do tempo precisou ser repensado. Optou-se por iniciar o App do Combustível, porque neste

aplicativo se reforça o conceito e prática de variáveis e o comando de condição, além dos conceitos de operações matemáticas e lógicas.

No desenvolvimento deste App, percebeu-se o interesse dos alunos que possuem carros brasileiros, também que a diferença de preços do combustível no Brasil e Uruguai é bastante significativa com relação ao custo da gasolina. Como já afirmava Knowles, Holton e Swamson (2011), o aprendiz precisa conectar o que está aprendendo com o porquê e com sua aplicação prática na sua vida. Não houve dificuldade na questão de desenvolver o raciocínio matemático para inferir as fórmulas necessárias para os cálculos.

Constatou-se que dois alunos se apropriaram dos conceitos e prática propostos no decorrer da aula. Outros três alunos, que demonstraram domínio operacional da ferramenta, copiaram os comandos, não sendo possível determinar a apropriação do conhecimento.

Na aula seguinte, o App Passos⁶⁶ foi proposto. Houve uma empolgação por parte dos alunos e foi realizada a programação da interface. Apenas um aluno não conseguiu desenvolver corretamente a interface. No encontro seguinte, a parte da programação (eventos) foi finalizada.

Verificou-se, pelas perguntas e pelo produto final, que dois alunos se apropriaram dos conhecimentos – conceitos e prática – envolvidos neste App. Dois alunos que não estavam na aula anterior desenvolveram a interface do aplicativo.

No encontro seguinte, terminaram a programação já iniciada. Percebendo que os alunos já não apresentavam dificuldades em desenvolver interfaces e também por uma questão de tempo, foi disponibilizado para a Calculadora, na Galeria⁶⁷ do *App Inventor*, o projeto da interface. Neste aplicativo foram trabalhadas operações matemáticas e dois tipos de laços de repetição: *for* (para) e *while* (enquanto).

Na 10ª aula, foi apresentado aos alunos o Pós-teste (ou Teste final), na versão da Língua Portuguesa e Espanhol. Para que a habilidade Compreensão do problema não fosse comprometida pelo entendimento (e tradução) do idioma, cada aluno escolheu em qual formato gostaria de respondê-lo. Todos optaram pelo seu idioma materno⁶⁸. Também nesta aula, o projeto final foi explanado. Foi discutido o tema do trabalho, o interesse para sua profissão, a forma de avaliação, o tempo para desenvolvimento entre todos os sujeitos presentes. Então, foram redigidas as especificações do projeto final.

⁶⁶ App que conta o número de passos e calcula a quantidade de metros percorrida, o tamanho dos passos e as calorias gastas.

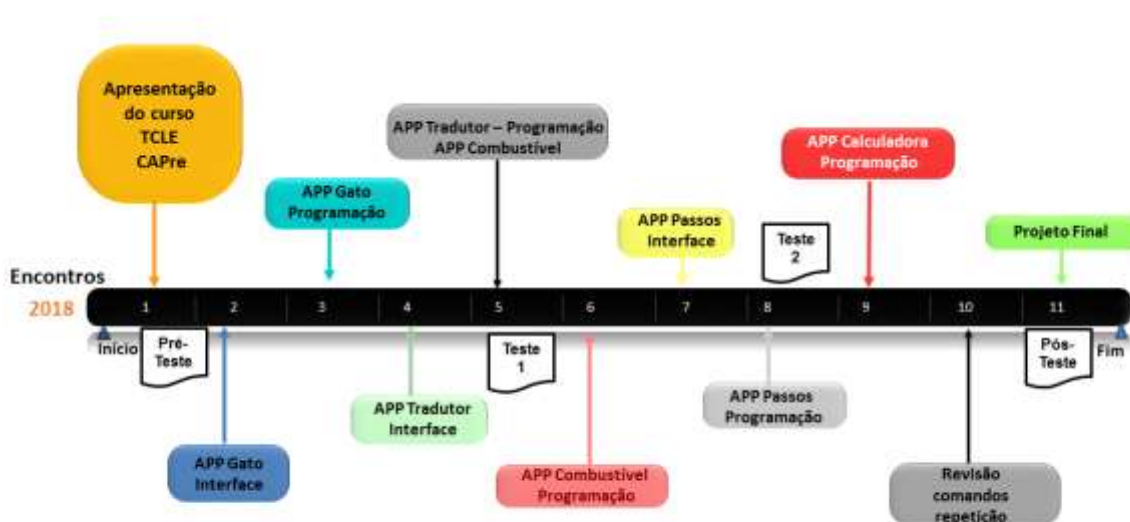
⁶⁷ Repositório no qual se compartilha com os usuários do *App Inventor* o projeto do App.

⁶⁸ Idioma falado no seu país de origem, também denominado de L1.

Esse projeto consistiu em realizar um cálculo simplificado de fretes rodoviários. Este documento está disponibilizado no AVA, no qual todos os dados referentes a este projeto foram descritos. Está disposto, inclusive, um exemplo de como realizar os cálculos. Um exemplo de interface foi disponibilizado para os alunos no AVA e na Galeria, porque o tempo que os alunos dispunham para desenvolver e apresentar o aplicativo se julgou insuficiente, considerando as demais atividades de final de semestre.

O último encontro foi voltado para o entendimento do projeto final, com a presença de 5 alunos. Ou seja, a Habilidade Compreensão foi uma das mais trabalhadas. Acredita-se que a maior dificuldade dos indivíduos ainda reside na Compreensão, ou seja, ler um problema e entender o que está escrito, já que sem esta habilidade todas as demais ficam comprometidas. Ademais, essa ideia pode ser comprovada porque, após a explicação, o debate, a interação com a professora e entre os colegas, os alunos presentes conseguiram entender o problema do frete a ser resolvido. Para as demais habilidades, como abstração, elas foram sendo empregadas e desenvolvidas de forma a perceber a sua aprendizagem. Algumas dúvidas surgiram na leitura de variáveis, no sentido de confirmar seu conhecimento e a corretude do que deveriam programar. A partir dessas perguntas percebeu-se que a habilidade de Resolução Algorítmica havia sido entendida. O resumo dos encontros está apresentado na Figura 30.

Figura 30 - *Timeline* das atividades do curso



Fonte: Da autora (2019).

É importante observar que na primeira aula foi solicitado que os alunos respondessem ao questionário “Conhecendo Você”, no quinto encontro foi aplicado o Teste 01 e no 8º encontro o Teste 02. Ambos os Testes foram disponibilizados através das ferramentas de comunicação ao grupo, ou seja, no AVA: Fórum e Tarefas e no grupo de *WhatsApp*.

Conforme pode ser verificado no documento Plano de Aulas, o App SMS não foi apresentado para os alunos, por motivo de tempo. Foi desenvolvido o App Fatorial para demonstrar e reforçar os comandos de repetição. Nesse sentido também foi disponibilizado um aplicativo denominado Cambio. Neste, a programação de comandos de condição, operadores matemáticos e lógicos foram enfatizados e apresentados na aula 10.

Na Figura 31 é mostrada uma das aulas do curso que foi desenvolvida no corredor da *Universidad*, pois houve um problema no laboratório e os alunos trabalharam com *tablets* e seus computadores pessoais.

Figura 31 – Um encontro no EC1



Fonte: Da autora (2018).

Para concluir a programação do projeto final, um tempo foi estimado e combinado com os alunos, que postaram o arquivo correspondente no ambiente virtual. Um aluno enviou por este meio, os demais enviaram por e-mail para a professora. A seguir, os resultados são apresentados e discutidos.

5.1 Resultados e discussão

O grupo inicial contou com 10 alunos, sendo dois brasileiros. O primeiro contato para entender o contexto dos alunos foi através das perguntas do questionário “Conhecendo Você”. Somente nove alunos responderam às questões, embora o mesmo tenha ficado disponível no AVA durante todo o curso.

A resposta foi unânime sobre o conhecimento de que não sabiam programar. Porém apenas dois alunos não souberam responder sobre o que é um algoritmo e somente duas respostas foram parcialmente completas.

Quando perguntados sobre a questão 03 (o que são variáveis e comandos de condição e repetição), 44% não responderam. Dois alunos relacionaram o conceito de variáveis com a resolução de equações matemáticas, evidenciada na frase: “*variable: símbolo o valor numérico que conforma una ecuación matemática*”. Os demais não responderam corretamente. Somente um aluno respondeu todas as perguntas dessa questão, o que aponta que não há subsunções para os conhecimentos sobre a habilidade Resolução Algorítmica.

Na questão seguinte, quando indagados sobre “programar ou ser programado”, mais da metade dos alunos concordaram, complementando que “*saber programar para adaptarse a las condiciones y exigencias del siglo 21, y no ser siempre programados*”, escreveu um aluno. Por mais que o grupo tenha dividido suas opiniões sobre esta questão, 100% dos alunos concordam que o mercado está exigindo este conhecimento cada vez mais. Também são unânimes com relação à expectativa do curso: aprender, visto a importância que demanda este conhecimento no mercado atual, conforme pode-se corroborar com as conclusões dos alunos:

Poder aprender más sobre el ámbito informático y adquirir conocimientos que me aporten en el ámbito laboral.

Adquirir conocimiento de fazer aplicativos úteis para todo tipo de usuário.

Minha expectativa é simplesmente aprender o máximo de programação possível e refletir sobre sua importância atualmente.

A partir dessas afirmações é possível considerar que os alunos entenderam a importância e objetivo do curso proposto.

Durante a realização das aulas, sempre se solicitou que os alunos enviassem o projeto dos seus aplicativos, mesmo que não estivessem completos ou com erros. Todos os projetos enviados foram corrigidos e retornados aos alunos com as devidas observações e correções no próprio código.

Com relação às avaliações, a

Tabela 2 mostra o resultado dos acertos por aluno relativos às habilidades desenvolvidas. Os sujeitos estão identificados pela letra A seguido de um número.

Tabela 2 - Resultado dos testes da avaliação objetiva por habilidades – EC1

ID aluno	H1/5	H2/6	H3/6	H4/4	H5/4	Total de acertos/9	Valor Final
A02	3	4	3	1	4	7	77,78
A03	3	3	2	0	0	5	55,56
A04	3	2	1	0	0	3	33,33
A06	3	4	4	1	1	5	55,56
A07	3	3	1	0	0	3	33,33
A08	3	3	4	1	1	6	66,67
A09	4	4	4	1	1	6	66,67
A10	1	2	1	0	0	3	33,33

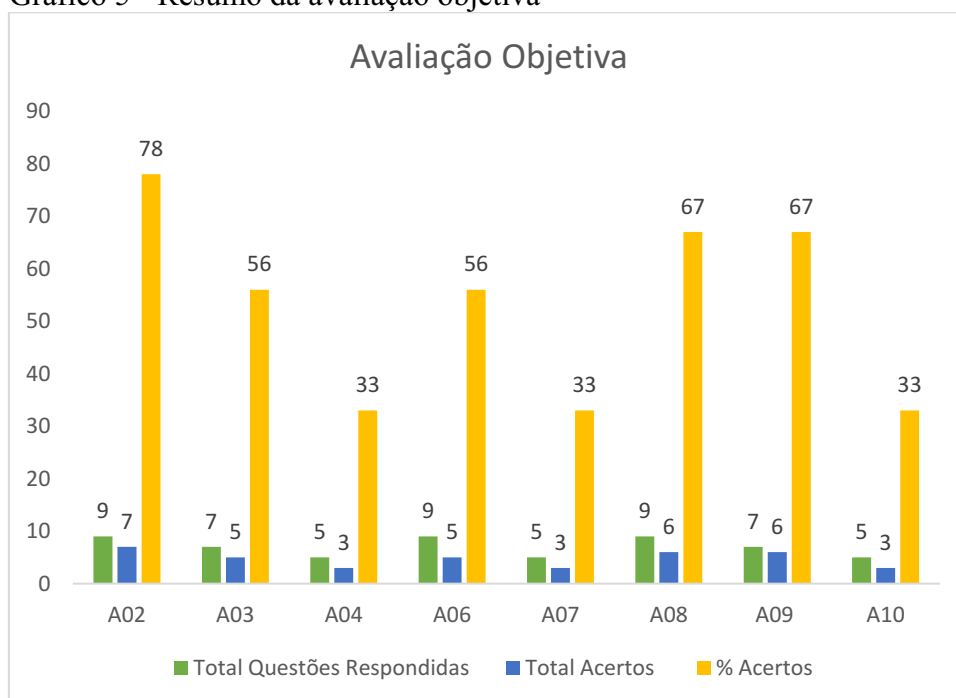
Fonte: Da autora (2019).

Conforme se observa na

Tabela 2, foram realizados três testes. A coluna H1/5 indica que em cinco questões (do total de 9) foram avaliadas a habilidade 1 – compreensão. E assim sucessivamente nas demais habilidades, dessa forma H2 é abstração, H3 é resolução de problemas, H4 é a habilidade de resolução algorítmica e na sucessora – H5 –Avaliação. Na coluna Total, está totalizado o número de questões acertadas pelo aluno e seu correspondente em percentual na seguinte coluna. No Apêndice F está mostrada a correção das questões dos testes por alunos.

Acompanhando a Tabela, os alunos 01 e 05 foram desconsiderados, pois ambos desistiram de acompanhar este curso e também da disciplina curricular de Informática. Alguns participantes não atingiram o percentual mínimo de 60% nesta parte da avaliação objetiva, porém não foram desconsiderados na avaliação subjetiva. Com a intenção de melhor ilustrar estes resultados, o Gráfico 5 foi construído.

Gráfico 5 - Resumo da avaliação objetiva



Fonte: Da autora (2019).

No Gráfico 5, pode-se perceber nas respostas dos alunos A04, A07 e A10, os quais responderam a somente cinco questões, todos acertaram apenas 3, e isso significa que participaram somente do pós-teste. Portanto, é possível inferir que os respondentes não haviam participado de testes anteriores, bem como não acertaram as questões 4 e 5 (mais complexas). No entanto, apesar desses resultados, 3 alunos obtiveram mais de 60% de acertos.

Quanto aos resultados do Teste 01, dos 8 alunos que enviaram, 50% acertaram a pergunta 1 e o mesmo resultado para a questão 02. Ou seja, somente dois alunos acertaram as duas perguntas.

Com relação ao Teste 02, somente 04 alunos o entregaram. Ao perceber-se que os alunos demoravam para retornar o teste, fez-se uma mediação com três alunos que estavam presentes. O resultado foi de que não haviam entendido (Habilidade Compreensão) a primeira questão, por isso não haviam enviado os resultados. Após, os alunos que participaram da intervenção e mais um (total de 4) enviaram os resultados, com 75% de acertos. Acredita-se que o aluno que enviou a resposta equivocada não havia participado da mediação. Conclui-se que a habilidade Compreensão é muito importante, pois eles não tinham resolvido o exercício proposto porque não conseguiram entender o que leram, ou seja, o que deveria ser resolvido. E mesmo com a questão sendo proposta no idioma escolhido, com o enunciado trazendo um exemplo, não foram capazes de estender o raciocínio lógico para obter a solução.

Na segunda questão do Teste 02, somente 25% dos alunos obtiveram êxito. Um aluno manifestou novamente que não havia compreendido o problema. Novamente se valida a importância da habilidade Compreensão. Os resultados do Pós-teste, com o total de 8 participantes, mostraram que com questões menos complexas (como, por exemplo, a primeira e segunda) os alunos acertaram 100%. À medida que a complexidade da questão aumenta, o índice de acertos diminui. Na questão 3, que avalia todas as habilidades de PC, houve 50% de acertos, em contraponto à questão 4, na qual não houve acertos. Na questão 5, 62,5% das respostas foram que não entenderam o que estava sendo solicitado. No que se refere aos alunos que tentaram respondê-la, não obtiveram êxito.

Nas duas últimas questões, houve a presença de uma porção de código. Não se pode desconsiderar a hipótese de que os alunos não acertaram, pois ainda não haviam sido desafiados com esse tipo de problema. Ressalta-se aqui que, em cursos futuros, este processo de pensamento algoritmo e avaliação precisa ser apresentado anteriormente ao pós-teste.

A Tabela 3 mostra o resultado já apresentado referente aos testes, acrescido da avaliação dos indicadores referentes ao projeto final, a fim de consolidar a avaliação objetiva.

Tabela 3 - Resultado da ACod (EC1)

ID aluno	Testes	Quantidade indicadores ProjF (9)	ProjF	Média	Parecer
A02	55,6	7	78	66,8	Indícios que está Acod.
A03	55,6	Não enviou			Sem o projeto final, não foi possível avaliar a alfabetização em código.
A04	33,3	4	57	45,2	Não foi suficiente para avaliar a alfabetização em código.
A06	55,56	7	78	66,8	Indícios que está Acod. Embora o percentual de acertos nos testes não chegou ao mínimo esperado, demonstrou conhecimentos no projeto final.
A07	33,3	Não enviou			Sem o projeto final, não foi possível avaliar a alfabetização em código.
A08	66,67	6	67	66,8	Indícios que está Acod.
A09	66,67	7	78	72,3	Indícios que está Acod.
A10	33,33	4	44	38,7	Não foi suficiente para determinar a alfabetização em código.

Fonte: Da autora (2019).

A partir dos dados apresentados na Tabela 3, no que tange aos alunos que não enviaram o projeto final, não foi possível a avaliação e, portanto, a afirmação de ser ou não ACod, ou

seja, não se pode julgar seus conhecimentos por falta de indicadores necessários. No Apêndice G está detalhada a correção do produto final segundo os indicadores definidos (Figura 27). Com respeito à avaliação dos alunos 04 e 10, não conseguiram atingir o mínimo de conhecimento; portanto, não foram considerados ACod. No total, 4 alunos apresentaram indícios de serem ACod.

Os alunos manifestaram que estavam no final do semestre letivo e finalizando muitas atividades de outras disciplinas, sendo assim um protótipo de interface foi desenvolvida pela autora e disponibilizada aos alunos. Corrobora-se com a posição seguinte:

A andragogia ressalta que os adultos tendem a ter grande número de preocupações e de problemas a resolver fora da situação de aprendizagem. Assim, as demandas da experiência de aprendizagem não devem ser irreais, deve haver um balanceamento adequado entre o tempo necessário para apresentação da situação de aprendizagem e o tempo necessário para a obtenção da aprendizagem (SANTOS, 2010, texto digital).

Considerando os 6 sujeitos que enviaram o Projeto Final, 66,6% dos App obtiveram nota superior a 60; portanto, pode-se responder positivamente à hipótese “c” da tese, na qual inquire se o sujeito pode desenvolver um projeto de *software* utilizando o PC.

Novamente, percebeu-se que a maior dificuldade está no primeiro pilar: compreensão. E apresenta-se a seguinte dúvida: como melhorar o processo de compreensão do indivíduo?

Já na Tabela 4, apresenta-se o resumo da segunda parte da avaliação, ou seja, a subjetiva, considerando ainda os alunos mostradas na Tabela 3, anterior.

Tabela 4 - Avaliação subjetiva

ID aluno	Avaliação Objetiva Média	App enviadas /4	CoP	Observação
A02	66,8	3	Sim, com contribuições importantes	Embora faltou 3 aulas, fazia as tarefas, seguia os tutoriais.
A03		0	Poucas intervenções	Três ausências, não conseguia realizar as atividades sozinha.
A04	45,17	2	Sim, com contribuições	Desenvolia e acompanhava bem as atividades da aula.
A06	66,8	3	Pouca interação. Seu contato mais no privado	Desenvolia e acompanhava bem as atividades da aula, e nas faltas conseguia acompanhar os tutoriais.

ID aluno	Avaliação Objetiva Média	App enviadas /4	CoP	Observação
A07		0	Pouca interação.	Desenvolvia e acompanhava bem as atividades da aula, com certa independência.
A08	67,00	2	Sim, com contribuições importantes	Desenvolvia e acompanhava bem as atividades da aula.
A09	72,3	4	Sim, com contribuições importantes	Desenvolvia e acompanhava bem as atividades da aula, e nas faltas conseguia acompanhar os tutoriais.
A10	38,7	3	Não participou. Somente uma vez em privado.	Não conseguia acompanhar as atividades da aula. Os App enviados sempre inacabados.

Fonte: Da autora (2019).

A partir dos dados apresentados na Tabela 4, pode-se acompanhar o desenvolvimento dos alunos quanto às atividades propostas durante o curso. Importante ressaltar que os sujeitos A02, A06, A08 e A09, que são aqueles considerados ACod, a partir da avaliação objetiva, apresentam um padrão na avaliação subjetiva. Em outras palavras, interagiram na comunidade de prática, tanto lançando dúvidas, quanto respondendo a questionamentos dos colegas. Ainda, desenvolveram os *App*, tanto em aula, quanto sozinhos (autodidatas) e compartilharam com o grupo.

Ao analisar o A03, pode-se perceber que faltou a aulas, embora tenha pontuado no último teste não conseguiu acompanhar o grupo, não se utilizou da CoP como ajuda, nem tampouco realizou o projeto final. Portanto, nada se pode afirmar acerca de que se houve melhora ou uma construção quanto ao desenvolvimento do PC, pois sua participação e desempenho não foram suficientes. Semelhante análise pode ser feita sobre o A10, que, embora tenha desenvolvido e entregue o projeto final, os resultados foram insuficientes nos indicadores, ou seja, não desenvolveu sequer a metade do que foi especificado no projeto. Sendo assim, determina-se conclusão análoga para o aluno A07.

Já ao analisar o A04, não atingiu ACod, porém desenvolveu bem os aplicativos da aula, embora somente entregou 2, percebeu-se um interesse no tema e nas três primeiras habilidades

testadas obteve êxito. Pode-se afirmar que este participante reúne indícios do desenvolvimento de PC, porém de forma desplugada (PCD).

No Quadro 8 está exibido o resumo de cada App enviado, organizado por aluno.

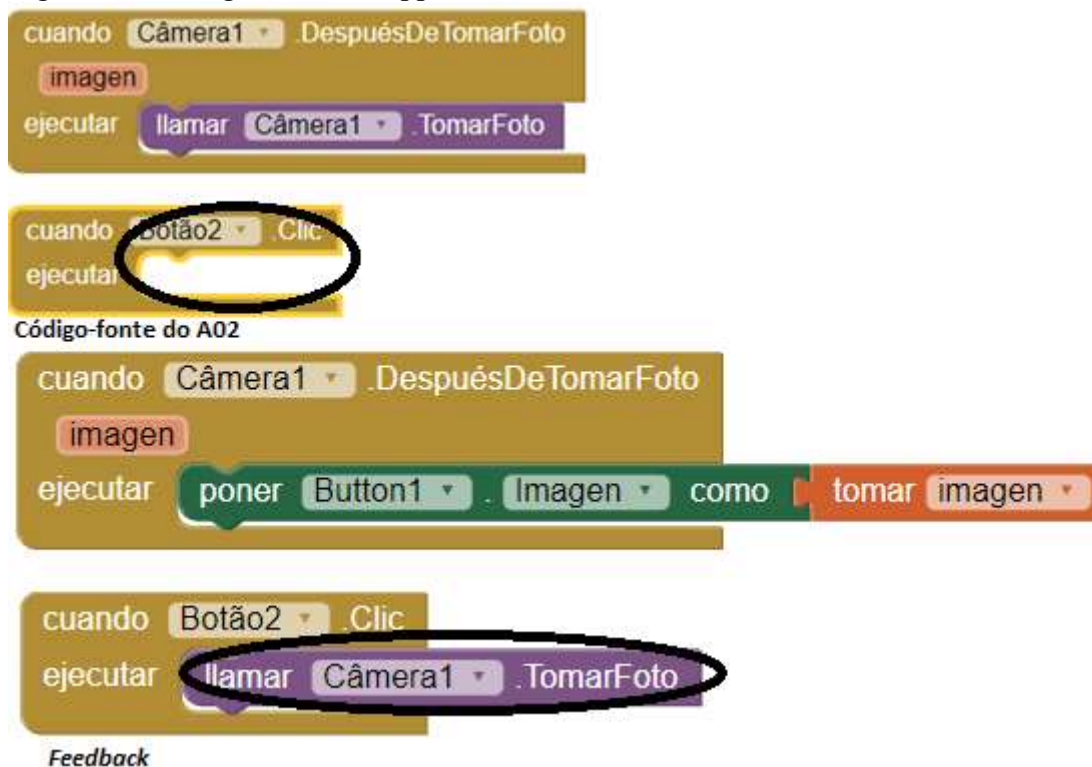
Quadro 8 - Resumo da avaliação dos App do EC1

ID Aluno	Gato	Tradutor	Gasolina	Passos
A02	Faltou no código-fonte a correta posição da foto.	Programação correta.	Código-fonte incompleto.	Não disponibilizou.
A04	Programação correta.	Enviou somente a interface, não há código-fonte.	Não disponibilizou.	Não disponibilizou.
A06	Programação correta.	Não disponibilizou.	Enviou somente a interface, não há código-fonte.	Código-fonte correto, sem erros.
A08	Programação correta.	Código-fonte completo.	Não disponibilizou.	Não disponibilizou.
A09	Código-fonte incompleto.	Sem programação, somente a interface.	Código-fonte correto, faltou o botão Limpar.	Interface completa e modificada. Código-fonte incompleto.
A10	Código-fonte incompleto. Interface definida e correta.	Código-fonte incompleto. Interface faltando componentes.	Código-fonte incompleto. Interface definida e correta.	Não disponibilizou.

Fonte: Da autora (2020).

A partir dos dados apresentados no Quadro anterior, uma análise de cada sujeito é percorrida. O aprendiz A02, conforme já comentando (Tabela 4), na comunidade de prática (CoP) fez contribuições importantes e, além disso, quando recebia o *feedback* da professora relacionado ao App enviado, procurou reescrever o código. A Figura 32, mostra uma parte do código-fonte desenvolvido pelo sujeito e o *feedback* correspondente referente ao aplicativo do Gato.

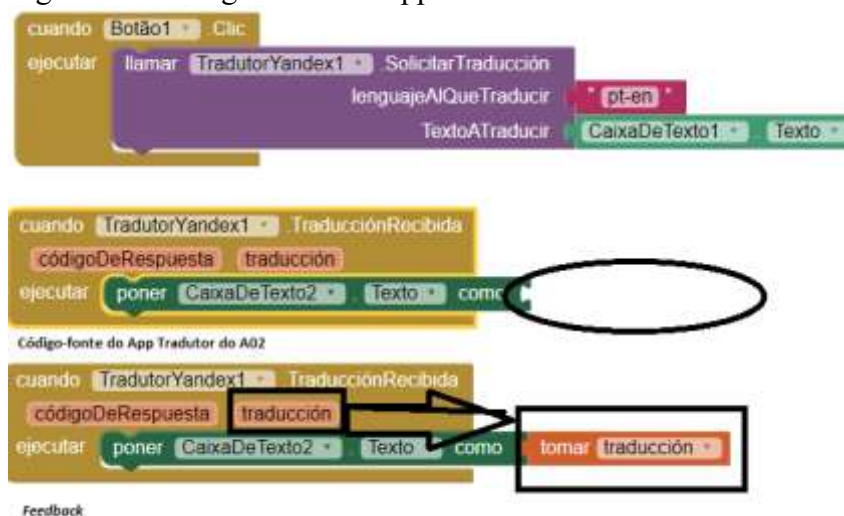
Figura 32 - Código-fonte do App Gato do A02



Fonte: Da autora (2020).

No código-fonte mostrado na Figura anterior, conclui-se que faltou complementar apenas um comando, conforme circulado. Já na Figura 33 está demonstrado um outro exemplo do feedback para o App Tradutor, para o mesmo sujeito.

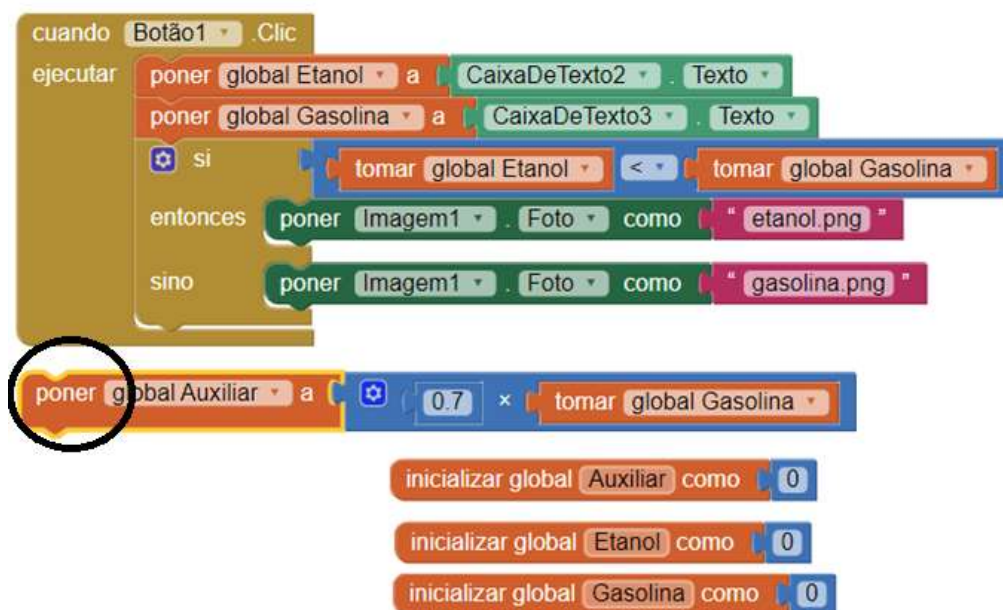
Figura 33 - Código-fonte do App Tradutor do A02



Fonte: Da autora (2020).

Como no exemplo anterior, ao A02 faltou somente incluir no código-fonte a terminação do comando, ou seja, arrastar o parâmetro “traducción” para o lugar correto, demonstrado na Figura anterior. Para o mesmo sujeito, no App Combustível, o código-fonte foi enviado faltando muitos comandos, conforme pode ser visualizado na Figura 34.

Figura 34 - Código-fonte do App Combustível do A02

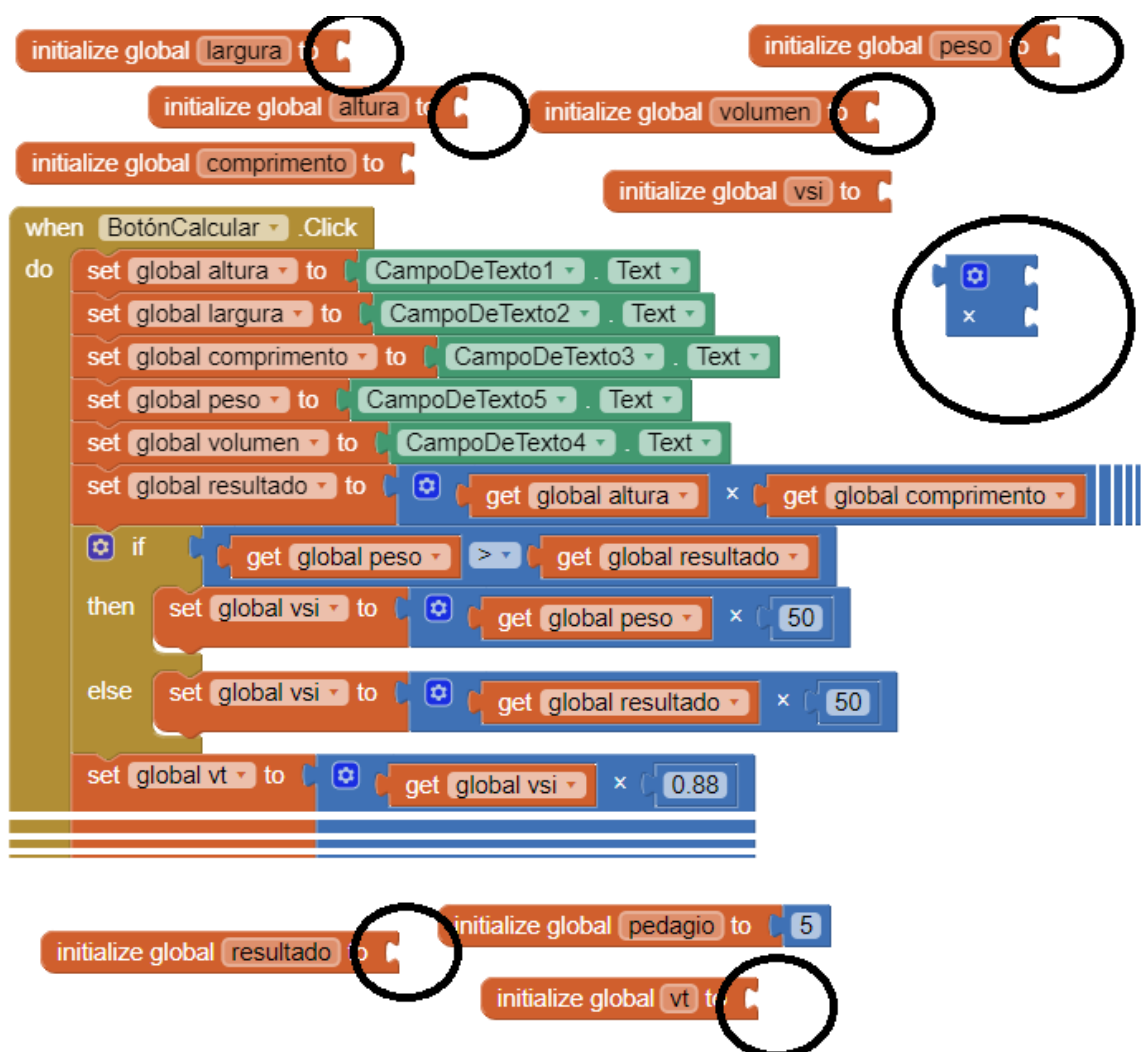


Fonte: Da autora (2020).

Na Figura anterior pode-se perceber o código inacabado, como também os comandos desconectados dos blocos (devidamente marcado na Figura 34).

Ao ser analisado os App do aluno A04, pode-se afirmar que o primeiro (Gato) seguiu as orientações da aula e o segundo não continha nenhum bloco de comando e por isso não serão apresentados. Já uma porção do código-fonte do projeto final pode ser visualizado na Figura 35.

Figura 35- Código-fonte do Projeto Final de A04

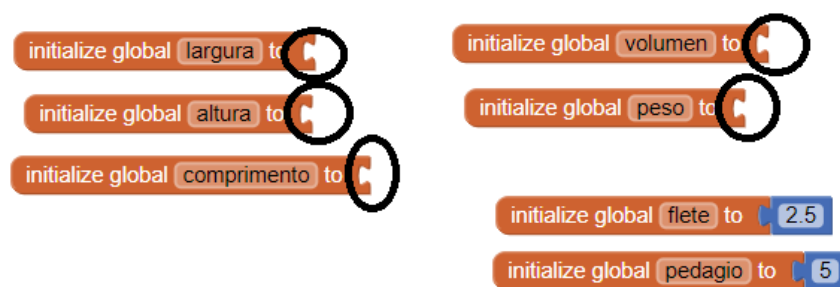


Fonte: Da autora (2020).

Ao ser avaliado o código-fonte apresentado na Figura precedente, pode-se apontar as diversas variáveis sem as devidas inicializações e ainda comandos desconectados de um bloco. Embora o App possa funcionar, a organização do código precisa ser melhorada.

O sujeito A06 enviou somente 2 App com os respectivos código-fonte iguais aos do tutorial, portanto não há razão para ser apresentado. Porém, a porção do código-fonte do projeto final deste aluno está descrita na Figura 36.

Figura 36 - Código-fonte do Projeto Final do A06



Fonte: Da autora (2020).

Pode-se inferir uma conclusão semelhante com respeito ao código-fonte do A04, para o aluno A06, no que tange à inicialização das variáveis, anotadas na Figura anterior. O restante do código-fonte está correto sintaticamente.

Analogamente, o aluno A08 enviou dois aplicativos com a programação completa e semelhante ao proposto no tutorial. Uma porção do código-fonte do projeto final pode ser observado na Figura 37 para este sujeito.

Figura 37 - Código-fonte do Projeto Final do A08



Fonte: Da autora (2020).

Mais uma vez, se apresenta uma parte do código-fonte com erros de sintaxe. Embora o *App Inventor* não o reconheça e a aplicação seja executada, percebe-se em vários aprendentes o mesmo erro.

O aluno A09, considerado ACod, apresentou o código-fonte mais completo com relação aos demais sujeitos da pesquisa e este pode ser visualizado na Figura 38.

Figura 38 - Código-fonte do Projeto Final do A09

```

initialize global vt to 0
initialize global Largura to 0
initialize global altura to 0
initialize global Comprimento to 0
initialize global Peso to 0
initialize global vsi to 0
initialize global Resultado to 0
initialize global Volumen to 0

when BotónCalcular .Click
do
  set global altura to CampoDeTexto1 . Text
  set global Largura to CampoDeTexto2 . Text
  set global Comprimento to CampoDeTexto3 . Text
  set global Peso to CampoDeTexto5 . Text
  set global Volumen to CampoDeTexto4 . Text
  set global Resultado to (get global altura) × (get global Largura)
  if (get global Peso) > (get global Resultado)
  then
    set global vsi to (get global Peso) × 50
  else
    set global vsi to (get global Resultado) × 50
  set global vt to (get global vsi) × 0.88
  set CampoDeTextototalsemimposto . Text to (get global vsi)
  set CampoDeTextototalcomimposto . Text to (get global vt)
  (get global Resultado) × (get global Comprimento) × (get global Volumen)
  
```

Fonte: Da autora (2020).

No código do projeto final (Figura anterior), conclui-se que faltou a soma na variável “vt” ademais que as variáveis estão todas inicializadas, demonstrando que o aluno obteve uma aprendizagem significativa.

E, por fim, na Figura 39 estão explicitados os códigos-fonte enviados pelo A10.

Figura 39 - Código-fonte do A10

```

when Som1 .SoundError
  message
do
  call Som1 .Play
  call Som1 .Vibrate
  millisecs 500

```

Código-fonte do App Gato

```

when Botão1 .Click
do
  call TradutorYandex1 .RequestTranslation
  languageToTranslate
  textToTranslate

```

Código-fonte do App Tradutor

```

when Botão1 .Click
do
  initialize local Etanol to
  in
  initialize local Gasolina to
  in

```

```

initialize local nome to
in

```

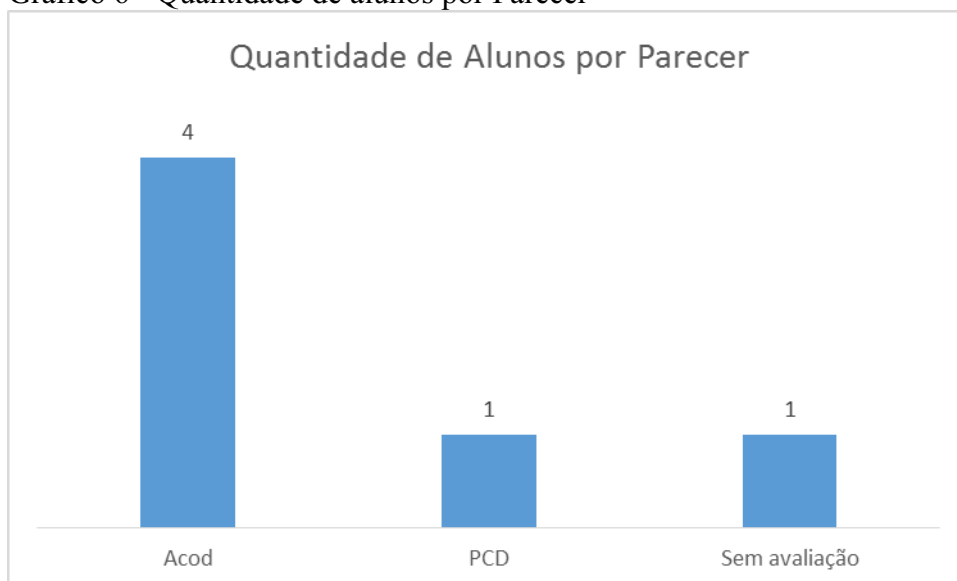
Código-fonte do App Combustível

Fonte: Da autora (2020).

Nesta Figura é imperativo ressaltar que os códigos mostrados para cada App enviado por A10 não são porções e, sim, o código-fonte completo do App respectivo. Após a análise dos mesmos e seguindo o MRPC, pode-se afirmar que este aluno não conseguiu formar subsunçores e, portanto, não conseguiu atingir o desenvolvimento do PC.

Em suma, percebe-se que a união das avaliações – objetiva e subjetiva – se complementam, inclusive para validar os alunos considerados alfabetizados em código. O resumo pode ser visto no Gráfico 6.

Gráfico 6 - Quantidade de alunos por Parecer



Fonte: Da autora (2020).

Nesse Gráfico, é apresentado o número de alunos que foram classificados como ACod, no caso, 4. Deste número, apenas um aluno fez sua formação no Brasil. O participante que desenvolveu o PC de forma desplugada teve sua formação no Uruguai e o restante não foi possível fazer avaliação.

Outro ponto importante percebido para compor a avaliação são os App. Estes pequenos aplicativos desenvolvidos em aula, que no final são enviados para a CoP, avaliados pelo professor que retorna sempre um *feedback*, é importante para o aprendizado e, conseqüentemente, para o desenvolvimento das habilidades consideradas nesta tese. Os alunos que atingiram a ACod, pode-se validar que entregaram a maioria dos App. Assim sendo, isso evidencia a importância dessa prática no processo de aprendizagem.

Iniciaram 10 alunos o curso, sendo que somente 7 chegaram no final. Deste grupo, 4 alunos (um brasileiro) podem ser considerados ACod. Nenhum destes alunos declarou conhecimentos prévios de programação, mas conseguiram desenvolver os App mesmo quando não estavam em aula (somente através do material instrucional disponibilizado em aula). Somando-se a isso, eles interagiram na CoP através do envio do que estava sendo solicitado e demandando suas dúvidas. Conclui-se que a CoP, a partir do seu papel no processo avaliativo, é um elemento importante para avaliar as habilidades consideradas nesta tese sobre PC. Em síntese, os alunos que se comprometeram com as atividades propostas conseguiram a ACod.

Um dos fatores que impactou nos resultados das avaliações dos alunos foram as ausências. Dos 11 encontros realizados, a média de faltas dos alunos foram 3, ou seja, os alunos não assistiram a 30% dos encontros. Os alunos classificados como ACod, mesmo não estando

presentes em aula, conseguiam realizar as atividades. Os demais alunos não tentaram recuperar os conteúdos perdidos, pois a cada encontro era desenvolvida a interface ou a programação de um App. Dessa forma, o aluno já não acompanhava a aula, e cada problema proposto (App) foi planejado para retomar conceitos anteriores e avançar em um novo conceito.

Cabe ressaltar que os sujeitos não apresentavam, no início do curso, nenhum subsunçor sobre algoritmos, lógica e linguagem de programação. Isso denota que, pelo menos, a Habilidade 4 foi construída totalmente durante as atividades propostas no constructo andragógico da tese. Sendo assim, todos os itens de avaliação são importantes e complementares.

Outro fator que poderia ser reconsiderado está na quantidade de horas do curso. Acredita-se que com um acréscimo de 10% permitiria uma melhor interação e compartilhamento de saberes. Neste caso piloto foram dispendidas 20 horas e, conforme o Plano de Aula, o App SMS não pôde ser trabalhado.

Todos os alunos que participaram do curso com mais de 50% de assiduidade, independente de terem atingido a ACod, receberam certificação.

Nenhum aluno respondeu ao último questionário, sendo que este foi repensando para o próximo estudo de caso, de modo que os certificados sejam concedidos mediante, também, à resposta do questionário. Uma ponderação que se julgou importante anotar é a questão de que esses alunos, em sua maioria, trabalham o dia todo. Parece que sobrecarregá-los com outras tarefas não é o ideal, até porque podem não responder o que realmente pensam e, em consequência disso, acabam por opinar de uma forma superficial, por falta de tempo, somente para cumprir uma atividade.

Em suma, este caso piloto resultou que o constructo andragógico fosse aplicado e avaliado, bem como a aplicabilidade do MRPC. A classificação dos sujeitos foi testada através da avaliação objetiva e bem corroborada pela avaliação subjetiva, ou seja, as avaliações se complementaram e não houve discordância entre elas. As hipóteses da pesquisa “a” e “c” foram respondidas afirmativamente, pois os sujeitos que não apresentavam nenhum subsunçor foram classificados como ACod ou PCD, que significa que conseguiram ancorar novos conceitos de forma significativa e produzir o projeto final a partir desses conhecimentos.

No capítulo seguinte, o segundo estudo de caso está narrado.

6 ESTUDO DE CASO 2 (EC2)

No capítulo anterior foi apresentado o Caso Piloto (ou EC1). O segundo estudo de caso, que corresponde aos Passos 6 e 7 da Fase 2 (Figura 15) foi realizado no Instituto Federal Sul-rio-grandense (IFSUL), no Câmpus Santana do Livramento (CSL), Rio Grande do Sul, mais precisamente no primeiro semestre do curso superior de Análise de Desenvolvimento de Sistemas. Tal como o EC1, este curso também é binacional e possui duração de 8 semestres. Segundo seu próprio nome já expressa, é voltado para a área de computação e, por consequência, com foco em programação.

No Apêndice E é mostrada a cópia da autorização deste estudo assinada pelo diretor do CSL, que foi realizado no ano de 2019, mais precisamente no horário da disciplina de Matemática Discreta (MD), ministrada pela professora Vanessa Cardoso. A escolha teve o intuito de contemplar o item “4.5 Linguagens de programação e funções” do Plano de Ensino desta disciplina, que está localizada no primeiro semestre da matriz curricular do curso.

Na disciplina estavam matriculados 33 alunos inicialmente. Destes, somente 21 terminaram o semestre e o estudo de caso, cumprindo todas as atividades propostas. Destes infere-se que a turma é constituída de 47% brasileiros e 53% uruguaios; que a média de idade é de 22,04 anos (considerando que um sujeito respondeu como zero) e que 80,9% são homens e 19% são mulheres.

Neste EC2 foram realizados 10 encontros, embora no Plano de Aula tenham sido planejados somente 8, de uma hora e trinta minutos cada um, no Laboratório de Informática no CSL. Os computadores foram previamente preparados para que os *softwares* necessários estivessem disponíveis.

Considerando a peculiaridade do EC2, ou seja, os sujeitos estão matriculados em curso da área de Computação, a interação no primeiro encontro foi de conhecê-los, de apresentar o

objetivo do curso e sua importância. Foi solicitada a assinatura do TCLE, o qual foi assinado pelos 27 alunos presentes, juntamente com o CAPre. Também responderam ao questionário “Conhecendo Você”, no AVA. Todos os sujeitos já apresentaram subsunçores com relação ao uso do AVA; portanto, não foram necessários maiores esclarecimentos, sendo, então, um indicativo de que no grupo todos estão alfabetizados digitalmente. No final do encontro, um grupo no aplicativo *WhatsApp* já havia sido criado.

Para atender às características desses sujeitos, ao pré-teste foram agregadas três novas questões que se referem à Habilidade 4 (Resolução Algorítmica), todas retiradas do *Bebras*. Uma dessas questões está mostrada na Figura 40.

Figura 40 - Questão 4 do Pré-teste

4. Jogue os dados / Tira los dados		
<p>Después de la escuela, los jóvenes castores acostumbran jugar juntos. Para evitar peleas sobre adónde jugar, ellos tiran un dado normal de seis lados. La decisión es encontrada de según esta regla:</p> <p>Depois da escola, os jovens castores costumam brincar juntos. Para evitar brigas sobre onde jogar, eles jogam um dado normal de seis lados. A decisão é encontrada de acordo com esta regra:</p>		
	<p>IF o primeiro lance é maior que o segundo lance THEN vamos brincar na floresta ELSE IF o terceiro lance é menor do que o primeiro lance THEN vamos brincar no rio ELSE vamos brincar no campo esportivo</p>	<p>IF el primer tiro es mayor que el segundo tiro THEN vamos a jugar en la floresta ELSE IF el tercer tiro es menor que el primer tiro THEN vamos a jugar em el río ELSE vamos a jugar em el campo deportivo</p>

Pregunta: / Questão:

¿Qué secuencia de tiros enviará a los jóvenes castores al campo deportivo?

Qual sequência de lances mandará os jovens castores para o campo esportivo?



a) A

b) B

c) C

d) D

e) não entendi a questão / no entendí la pregunta

País proponente: Alemanha (B) – 2016

Habilidade: projeto e implementação de algoritmos

Na questão apresentada na Figura 40, se pode observar o uso do comando de condição, avaliando a habilidade de reconhecer um trecho de algoritmo e os operadores relacionais.

Na avaliação dos resultados do Caso Piloto (EC1), na seção 5.1, foi observado que nas duas últimas questões do pós-teste os referidos alunos apresentaram dificuldade nos problemas que envolveram interpretação de código. A fim de mitigar essa situação, o Teste 3 (disponível no AVA) foi totalmente desenvolvido para esse conjunto de aprendizes, pois estavam apresentando um bom desenvolvimento, em sua maioria, dos aplicativos propostos. Esse teste foi planejado para avaliar o conhecimento em código utilizando o *App Inventor*, e foram sugeridas 4 questões desenvolvidas pela autora. Em todas as questões, a Habilidade 1 (Compreensão) e 5 (Avaliação) foram observadas. Nas demais questões são mensuradas as Habilidades 3 (Resolução de Problemas) e 4 (Resolução Algorítmica).

No segundo encontro, foi apresentado o pré-teste, sendo respondido por 20 alunos presentes. Cabe salientar que a partir deste momento estão sendo considerados somente os sujeitos que concluíram o curso “Eu Programo 1.0!”, cumprindo com todas as tarefas solicitadas, no total de 21 sujeitos. O App do Gato foi proposto, e tanto a interface quanto a programação foram executadas. Os aprendizes não apresentaram dificuldade no uso do *App Inventor*. Devido ao fato de que foi a primeira vez que estavam programando para dispositivos móveis, os sujeitos ficaram realmente empolgados, pois o encontro que deveria ter sido finalizado às 22h15min se estendeu até às 23h. Os alunos desejaram finalizar o App e testá-lo. O teste com a foto pessoal não foi possível realizá-lo por falta de tempo. Retomando o princípio 6 (motivação para aprender, descrito na seção 2.3.2) de Knowles, Holton e Swanson (2011) e Ausubel (1983), que afirmam que o aprendiz precisa estar motivado para aprender, pode ser confirmado.

Nesse mesmo encontro, percebeu-se que alguns alunos já estavam alterando as propriedades dos objetos e solicitando que o próximo App já fosse disponibilizado para implementação. Desta afirmação, pode-se inferir a motivação existente e a aprendizagem por descoberta sendo concretizada. Desse modo, passou-se a analisar as respostas da questão “Você sabe programar?”, do questionário Conhecendo Você, na qual 16 respostas foram positivas, considerando o total de 21 sujeitos.

No encontro seguinte (3º), foram retomados os conhecimentos apresentados na aula anterior e o App Tradutor foi desenvolvido. Para os conceitos abordados neste aplicativo, como, por exemplo, variáveis e comandos de condições, os aprendizes não demonstraram dificuldade em implementá-los. A maioria dos alunos desenvolveu, concomitantemente, seus aplicativos com a professora durante a aula.

No quarto encontro, após a retomada da aula anterior, mais precisamente os subsunçores de condição e tipos de variável, observou-se que a maioria da turma havia ancorado os conceitos. Também se fez uma explanação sobre os tipos de combustível utilizados no Brasil (gasolina e álcool) e seus diferentes rendimentos de consumo. Uma solicitação unânime dos alunos foi para que a professora/tutora fizesse junto com eles, passo a passo, o desenvolvimento do App, para que todos acompanhassem a programação, inclusive, desenvolvendo a fórmula de cálculo exigida para o correto funcionamento do aplicativo. Mais de 50% dos aprendizes terminaram o App no período da aula, iniciando a próxima tarefa: o App dos Passos, para o qual também foi solicitado que a professora fizesse conjuntamente com os alunos o tutorial. Este último não foi concluído na aula.

Já no quinto encontro, os alunos solicitaram que desejavam seguir para a programação do próximo aplicativo, pois estavam motivados a aprender novos comandos. Ficou acordado que, para não prejudicar a avaliação do curso, os alunos iriam terminar o desenvolvimento do App dos Passos fora do horário das aulas e disponibilizariam no AVA, o que efetivamente aconteceu. Logo, a interface do App Calculadora foi disponibilizada na Galeria do *App Inventor* e a programação foi desenvolvida e findada. Interessante ressaltar que para o desenvolvimento do App Calculadora se usa o comando de repetição e a sua codificação em blocos gerou, em alguns alunos, uma certa dificuldade para implementá-lo.

Após essa atividade, foi aplicado o segundo teste, semelhante ao EC1.

Houve um feriado e no sexto encontro os sujeitos implementaram o App Fatorial, sendo que a maioria dos alunos o fez através do tutorial. Algumas intervenções foram realizadas no sentido de corrigir erros nos códigos. Percebeu-se que alguns sujeitos já estavam adicionando novos componentes e alterando suas propriedades. Isso aponta para uma aprendizagem por descoberta, segundo Ausubel (2003).

No encontro de número 7, foi proposto o teste 3. Após o término deste, foi realizada a sua correção e logo a seguir foi iniciada a implementação do App Cambio. Como neste aplicativo é exigida uma maior quantidade de cálculos matemáticos e comandos de condição, os alunos foram desafiados a desenvolver uma nova versão, na qual há a funcionalidade de escolher o tipo de moeda (reais ou pesos) para informar o troco.

Ao se aproximar do final do curso e da necessidade de desenvolver um projeto final, a professora responsável da disciplina sugeriu que os alunos desenvolvessem um jogo em formato de Quiz. Os sujeitos concordaram com a sugestão e os temas a serem abordados no jogo seriam o conteúdo já trabalhado na disciplina, a saber: permutação, arranjos e combinações. A docente disponibilizou uma lista de questões sobre esses temas. Aos alunos

foram distribuídas cinco questões para serem implementadas. Cada aluno, ao implementar a questão no jogo, precisou, previamente, resolver o problema e disponibilizar quatro alternativas para que o usuário selecione a correta. O propósito da atividade foi de conectar este curso com os conteúdos da disciplina de Matemática Discreta. Além disso, cada aluno foi designado como testador de dois aplicativos, programados por seus colegas, sendo que esta tarefa foi com o intuito de estudar os conteúdos matemáticos. Nesta atividade de teste, o aluno/testador deveria observar e avaliar os seguintes itens:

- a) **interface:** verificar se todos os elementos aparecem, se é amigável ao usuário e se a navegação está apropriada;
- b) **programação correta:** contador exato para o número de acertos e apresentação de todas as questões;
- c) **respostas:** resultado das questões está correto.

Com respeito ao curso “Eu Programa 1.0!”, foi necessário preparar um novo material – Aula 09 – a fim de apresentar os comandos necessários para implementar o Quiz. Como sugestão se apresentou o comando LIST.

Com relação ao interesse desse grupo formado por adultos, se recomenda que sejam propostos problemas da vida cotidiana (KNOWLES; HOLTON; SWAMSON, 2011). Acredita-se que o indivíduo, matriculado em um curso na área da Computação, será um programador e que, conseqüentemente, terá de desenvolver soluções para outras pessoas ou empresas. O fato do aplicativo final ser um jogo motivou os aprendentes, porque muitos sinalizaram que nunca haviam se aventurado nessa área.

Então, no oitavo encontro, foi implementado o conceito de listas, bem como outras possíveis soluções foram sugeridas, como, por exemplo, cada resposta pode ser atribuída a um componente do tipo botão. Um exemplo em aula foi desenvolvido a fim de experienciar os novos componentes. Não foi solicitado que este App fosse compartilhado entre os sujeitos, pois o final do semestre se aproximava e a prioridade ficou com a programação do projeto final. Por solicitação dos alunos, a professora Vanessa, no encontro seguinte, revisou os conceitos matemáticos necessários à resolução e implementação do Quiz. Houve uma participação intensa dos alunos que perceberam a importância dessa aprendizagem para a correta implementação do jogo de perguntas e respostas. Alguns sujeitos já iniciaram a programação do projeto final nesta aula. As Figura 41 e 42 mostram, em diferentes ângulos, o Laboratório de Informática, no 9º encontro.

Figura 41 - Alunos fazendo suas atividades



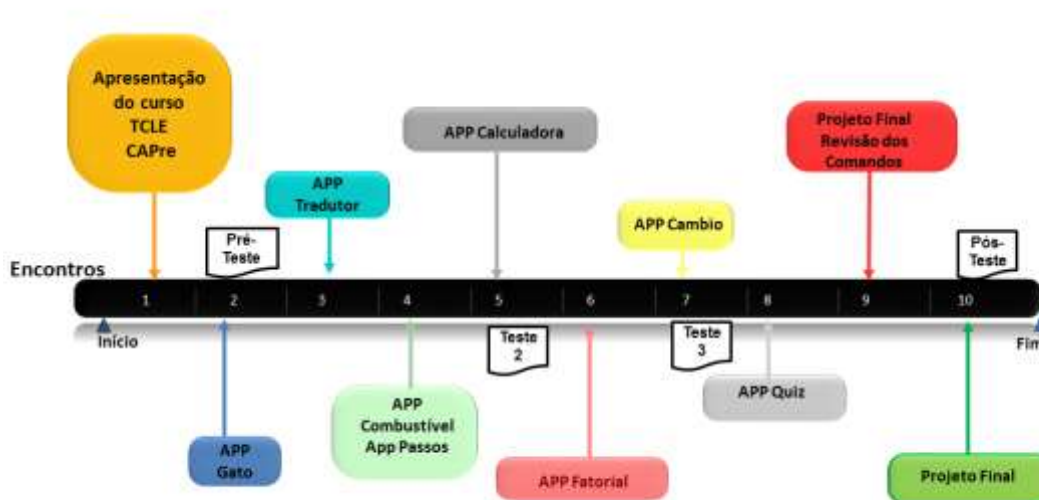
Fonte: Da autora (2019).

Figura 42 - Intervenção da professora



Fonte: Da autora (2019).

E, por fim, o último encontro, foi reservado para o desenvolvimento do projeto final, ou, pelo menos, parte dele, com o intuito de auxiliar os sujeitos na administração do seu tempo, pois os alunos estavam terminando o semestre, fato que pode ser traduzido com o acúmulo de trabalhos e provas, e, ademais, que nove alunos apontaram que trabalham durante o dia. Nesta aula, a professora Vanessa se fez presente (junto com a pesquisadora) à disposição dos alunos com o propósito de orientá-los em suas dúvidas. O resumo dos encontros está apresentado na Figura 43.

Figura 43 - *Timeline* das atividades do EC2

Fonte: Da autora (2020).

A partir da descrição do ocorrido em cada encontro, demonstrado graficamente na Figura 43, e dos testes realizados, desenvolve-se a seguir os resultados dos dados coletados neste estudo de caso.

6.1 Resultados e discussão

Neste EC2 foram envolvidos 21 sujeitos, os quais realizaram todas as atividades propostas no estudo. A partir dos dados coletados no questionário “Conhecendo Você”, ao qual somente 20 alunos responderam, sendo que 8 alunos (38,09%) declararam que trabalham e sete que o seu salário contribui para a renda familiar.

Quando questionados se sabiam programar, 76,19% (16) responderam que sim. Deste grupo, 12 sujeitos declararam que já haviam frequentado cursos de informática na área de programação em nível de Ensino Médio (que equivale ao *bachillerato* no Uruguai) ou subsequente. Do total de respostas do questionário, somente um aluno não definiu corretamente o conceito de algoritmos.

De acordo com os depoimentos registrados por parte dos sujeitos, a maioria deseja aprender a programar a fim de conquistar melhores trabalhos, conforme pode ser citado: “*Poder ter um bom emprego e aprender sobre a profissão que mais cresce no mundo*” e “*Mejorar mis conocimientos en el area y conseguir mejores oportunidades de trabajo*”. Pode-se perceber que entendem a importância sobre a programação (do PC) no contexto atual do mercado de trabalho.

Ficou evidente que todos os alunos, mesmo os que declararam que não sabiam programar, demonstraram conhecimentos básicos sobre conceitos de algoritmos e variáveis. Manteve-se a solicitação de que os aplicativos desenvolvidos fossem compartilhados com a turma no AVA assim que fossem finalizados ou no término do encontro. Sempre na aula posterior, para os App que estavam disponíveis, um *feedback* da pesquisadora era realizado. Os sujeitos que declararam que não sabiam programar, durante as aulas receberam uma atenção especial. Durante os encontros, alguns alunos procuraram a professora com dúvidas sobre o *feedback*, por exemplo, perguntando onde haviam errado ou o que poderiam fazer para melhorar o App. Isso demonstrou o interesse despertado e a motivação para aprender por parte dos sujeitos. Pode-se citar outra intervenção, na qual um grupo de alunos questionou a professora se era possível desenvolver outros aplicativos móveis, como uma rede de supermercados, por exemplo, e comercializá-los.

Ao longo dos encontros, percebeu-se que os aprendentes apresentaram novos interesses, e foi por esse motivo que um novo teste foi aplicado com o objetivo de verificar o conhecimento, mais especificamente na área de programação. Além disso, acrescentou-se a aula sobre desenvolvimento de questões para o Quiz (tema do projeto final). Também no pré-teste foram acrescentadas três novas questões. As alterações no plano inicial foram efetuadas com o intuito de conhecer melhor os alunos com relação à aprendizagem, visto que declararam conhecimento de programação. Remetendo ao MRPC, estes alunos seriam, desde o início, considerados ACod e, portanto, se fez indispensável analisá-los para verificar se efetivamente o são e se podem ser considerados LCod.

O resultado dos testes, segundo as habilidades, está mostrado na Tabela 5.

Tabela 5 - Resultado dos testes da avaliação objetiva por habilidade – EC2

ID aluno	H1/12	H2/6	H3/11	H4/10	H5/11	Total de acertos/16	Valor Final
A01	8	4	9	5	7	11	68,8
A02	8	4	9	6	7	11	68,8
A03	11	5	10	8	8	14	87,5
A04	8	3	10	6	8	12	75,0
A05	7	5	8	4	5	10	62,5
A06	9	3	8	7	7	11	68,8
A07	7	0	7	6	6	8	50,0
A08	9	3	7	7	7	11	68,8
A09	7	1	5	4	6	5	31,3
A10	10	4	9	7	7	12	75,0

ID aluno	H1/12	H2/6	H3/11	H4/10	H5/11	Total de acertos/16	Valor Final
A11	4	3	4	2	2	5	31,3
A12	7	3	7	5	6	9	56,3
A13	7	3	8	4	4	9	56,3
A14	11	4	9	8	8	13	81,3
A15	7	4	7	5	7	10	62,5
A16	10	4	10	8	9	13	81,3
A17	4	2	6	3	4	7	43,8
A18	9	4	8	6	7	12	75,0
A19	6	1	7	5	5	8	50,0
A20	10	4	9	7	8	13	81,3
A21	1	1	1	0	0	1	6,3

Fonte: Da autora (2020).

Conforme se pode observar na Tabela 5⁶⁹, foram considerados os quatro testes realizados, nesta ordem: pré-teste (ou teste 1), teste 2, teste 3 e pós-teste (teste 4). As colunas subsequentes apresentam o número de questões corretas do sujeito por habilidade, considerando que a coluna H1/12 indica a quantidade de acertos do sujeito na primeira habilidade (Compreensão), sobre o total de 12, e as demais colunas seguem de forma semelhante. Os acertos estão discriminados na coluna Acerto e o percentual correspondente na coluna mais à direita.

A fim de compreender os resultados, se acatou a declaração dos sujeitos sobre o conhecimento em programação. Portanto, os alunos A05, A07, A09, A11 e A19 afirmaram que não sabem programar, enquanto que os demais responderam que sim. O aluno A21 não respondeu à pergunta, sendo considerado como não alfabetizado em código. Deste grupo somente o A05 atingiu o percentual mínimo de acertos que foi de 62,5%.

Os sujeitos A07 e A21 somente realizaram a metade dos testes, sendo que A07 não estava presente nos encontros que foram realizados os testes 2 e pós-teste. Já o aluno A21 não respondeu ao pré-teste, nem ao teste 3. Considerando que estes dois testes perfazem um total de 10 questões, seu percentual de acertos ficou muito abaixo da média da turma que é de 61,1%. Ainda a ser ponderado sobre esse sujeito, que no teste 2, composto por duas questões, errou ambas, complementando que o total de acertos do pós-teste foi somente uma questão, do total de 5.

⁶⁹ No Apêndice H é mostrada esta Tabela acrescida dos dados de Total de Questões respondidas e o número de testes realizados pelo sujeito.

Analisando o sujeito A07, que respondeu ao pré-teste e ao teste 3, totalizando 9 questões, somente errou uma, ou seja, 88,9% de acertos sobre as questões respondidas. Ou seja, mesmo calculando o percentual sobre os testes propostos (em contrapartida com os testes respondidos pelos sujeitos), esse aluno (que declarou que não sabia programar) apresentou indícios de aprendizagem significativa para o PC.

Quanto ao sujeito A09 realizou todos os testes, sendo que em duas questões assinalou que não havia entendido o enunciado. Essa informação é importante anotar, pois há uma dificuldade por parte dos sujeitos em escolher esse tipo de resposta, ou seja, apenas três alunos o fizeram. Esse fato pode ser exemplificado na questão 02 do segundo teste⁷⁰. O aluno A04 respondeu que não entendeu a questão (NE), o A18 acertou e os demais todos erraram. Também pode ser observado na quarta pergunta do pós-teste, na qual não houve resposta NE e somente um sujeito acertou a questão. No mesmo teste, na questão 5, 63,15% dos respondentes marcaram respostas equivocadas.

O aluno A11 não respondeu ao teste 3; e das 12 questões respondidas somente acertou 5, perfazendo um total de 41,7%. Já o A19 não respondeu ao segundo teste, porém no pós-teste não obteve nenhuma resposta correta.

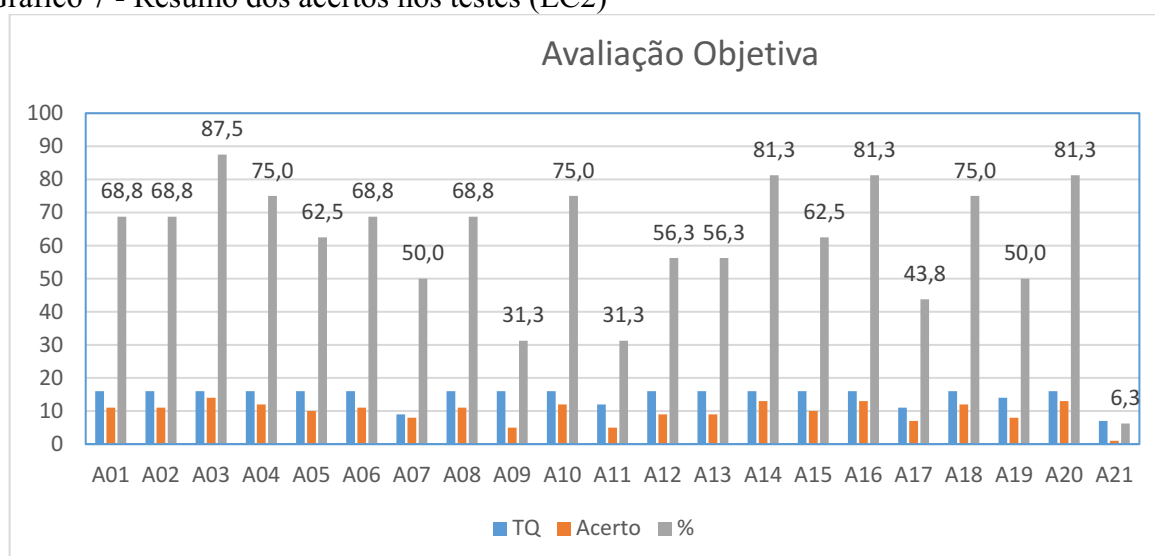
Com relação aos sujeitos A12, A13 e A17, declarados com conhecimento de programação, não atingiram os 60% necessários, mesmo participando de todos os testes propostos. Assim, para esses alunos a avaliação subjetiva torna-se importante.

A terceira pergunta do teste 3, que indaga qual o resultado para valores de entrada previamente definidos, foi a única questão que todos os alunos acertaram. Nesta é avaliada as habilidades de compreensão (H1) e avaliação (H5).

O resumo do desempenho dos sujeitos pode ser visualizado no Gráfico 7. O Apêndice I contém os dados de acertos de cada aluno, organizados por questão.

⁷⁰ Teste do Labirinto que avalia Pensamento Algorítmico (H04), resolução de problema (H3) e avaliação (H5).

Gráfico 7 - Resumo dos acertos nos testes (EC2)

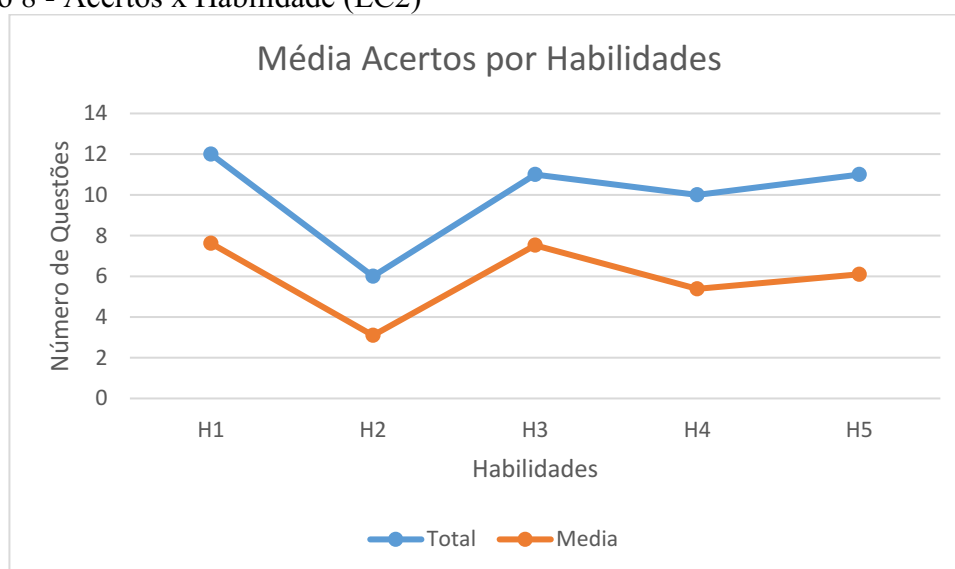


Fonte: Da autora (2020).

A partir da análise do Gráfico anterior, 61,9% dos alunos atingiram a nota mínima de 60, sendo este resultado considerado positivo para a investigação da aprendizagem do PC.

Já no Gráfico 8, é apresentada a média de acertos organizada por habilidade avaliada.

Gráfico 8 - Acertos x Habilidade (EC2)



Fonte: Da autora (2020).

A partir da observação do Gráfico 8, pode-se concluir que não houve uma diferença da quantidade de acertos nas habilidades propostas nesta tese, pois as linhas do gráfico variaram de forma semelhante.

Para completar a avaliação objetiva, o Projeto Final foi avaliado, seguindo o *template* proposto na Figura 27 do capítulo 4. O resumo da avaliação está demonstrado na Tabela 6.

Tabela 6 - Resultado da ACod (EC2)

ID aluno	Testes	QIA ⁷¹	Quantidade Indicadores atendidos	ProjF	Média	Parecer
A01	68,8	8,0	8,0	100,0	84,4	Acod
A02	68,8	9,0	8,0	88,9	78,8	Acod
A03	87,5	8,0	8,0	100,0	93,8	Acod
A04	75,0	9,0	4,0	44,4	59,7	Acod
A05	62,5	8,0	8,0	100,0	81,3	Acod
A06	68,8	8,0	3,0	37,5	53,1	Necessário observar as demais avaliações pois a diferença de valores nos testes e no produto final foi significativa.
A07	50,0	8,0	4,0	50,0	50,0	Não foi suficiente para avaliar a alfabetização em código. Declarou que não conhecia programação.
A08	68,8	8,0	8,0	100,0	84,4	Acod
A09	31,3	8,0	3,0	37,5	34,4	Não foi suficiente para avaliar a alfabetização em código. Declarou que não conhecia programação.
A10	75,0	8,0	7,0	87,5	81,3	ACod
A11	31,3	9,0	5	55,6	43,4	Não foi suficiente para avaliar a alfabetização em código. Declarou que não conhecia programação.

⁷¹ QIA: Quantidade de Indicadores Avaliados.

ID aluno	Testes	QIA ⁷¹	Quantidade Indicadores atendidos	ProjF	Média	Parecer
A12	56,3	9,0	9,0	100,0	78,1	ACod
A13	56,3	8,0	8,0	100,0	78,1	ACod
A14	81,3	8,0	8,0	100,0	90,6	ACod
A15	62,5	9,0	6,0	66,7	64,6	ACod
A16	81,3	8,0	7,0	87,5	84,4	ACod
A17	43,8	9,0	6,0	66,7	55,2	Não foi suficiente para avaliar a alfabetização em código. Declarou que conhecia programação.
A18	75,0	8,0	2,0	25,0	50,0	Não foi suficiente para avaliar a alfabetização em código. Declarou que conhecia programação.
A19	50,0	8,0	6,0	75,0	62,5	ACod
A20	81,3	9,0	7,0	77,8	79,5	ACod
A21	6,3	8,0	7	87,5	46,9	Não foi suficiente para avaliar a alfabetização em código. Não é conhecido seu estado anterior.

Fonte: Da autora (2020).

Faz-se necessário explicar a presença da coluna QIA. No MRPC são propostos vários itens para avaliação do produto (Figura 27), dentre eles a “Validação dos dados de entrada” (da característica Usabilidade e da subcaracterística “Proteção contra erros do usuário”). Então, dependendo de como foi programada a interface do aplicativo, não se torna necessário esta validação, por exemplo, no caso onde o usuário não digita nenhum dado de entrada. Portanto, esse item pode ser desconsiderado da avaliação. No Apêndice J está a avaliação do produto final de cada aluno, segundo as subcaracterísticas da norma ISO/IEC 25010, conforme o *template* proposto nesta tese.

A partir dos dados apresentados na Tabela 6, pode-se verificar que do grupo de alunos que declararam que não conheciam programação, que são A05, A07, A09, A11 e A19, os sujeitos A05 e A19 demonstraram o conhecimento necessário proposto no MRPC, sendo então considerados ACod. Outros, que declararam conhecimentos de programação, não foi possível demonstrar, através dos testes deste constructo andragógico, a assimilação dos conceitos, como, por exemplo, o A06, A17 e A18.

A seguir, na Tabela 7, apresenta-se o resumo da avaliação subjetiva, definida no MRPC, na qual são considerados os App enviados, a interação na CoP e as observações realizadas ao longo do curso.

Tabela 7 - Resultado da avaliação subjetiva do EC2

ID aluno	Avaliação Objetiva	App enviados/7	CoP	Observação
A01	84,4	7	Nenhuma interação	Realizou todas as tarefas, acompanhava o desenvolvimento dos App, tanto em aula quanto através dos tutoriais. Somente atrasou a entrega de um App.
A02	78,8	6	Nenhuma interação	Acompanhou o desenvolvimento dos App somente pelos tutoriais. Demonstrou grande interesse pelo código e linguagem. Já no segundo App (tradutor) desenvolveu seu código com estruturas condicionais aninhadas.
A03	93,8	7	Nenhuma interação	Desenvolia e acompanhava bem as atividades propostas na aula. Não apresentou dificuldade para implementar com o uso do tutorial. Cumpriu com a entrega de todos os App no prazo.
A04	59,7	5	Sim, com contribuições importantes	Desenvolia e acompanhava bem as atividades da aula e nas suas ausências programou seus App através do tutorial. Entregou um programa com atraso. Fez interações na CoP pertinentes. A partir da avaliação subjetiva, pode-se validar a avaliação objetiva, e, portanto, sua nota final objetiva pode ser arredondada para 6 (sujeito ACod).
A05	81,3	7	Pouca interação.	Desenvolia e acompanhava bem as atividades da aula e nas faltas desenvolvia o aplicativo se baseando no tutorial. Entregou um App com atraso.
A06	53,1	5	Pouca interação.	O sujeito não compareceu a vários encontros. Apresentou dificuldades em seguir o tutorial, frequentemente perguntando ao tutor sobre as tarefas realizadas em encontros anteriores.
A07	50,0	6	Nenhuma interação	Acompanhava as aulas, porém não conseguiu implementar um App através do tutorial, exceto até o App de Passos. Enviou um App com atraso.
A08	84,4	7	Nenhuma interação	Desenvolveu e acompanhou bem as atividades da aula e nas faltas desenvolvia o aplicativo através do tutorial. Entregou um App com atraso.
A09	34,4	6	Pouca interação.	Dificuldade de acompanhar as atividades da aula. Poucos App finalizados.
A10	81,3	7	Pouca interação.	Acompanhava as aulas, porém não conseguia acompanhar o tutorial, situação até o App dos Passos. Apresentou dificuldade de

ID aluno	Avaliação Objetiva	App enviados/7	CoP	Observação
				desenvolver e implementar fórmulas matemáticas (correspondente às habilidades H3 e H4). Bastante interativo em sala de aula.
A11	43,4	7	Nenhuma interação	Dificuldade de acompanhar as atividades da aula. Poucos App finalizados. Entregou um App Fatorial fora do prazo.
A12	78,1	7	Sim, com contribuições importantes	Desenvolia e acompanhava bem as atividades da aula, quando faltava desenvolvia os App através do tutorial. Durante os encontros interagiu muito com seus pares, bem como na CodP.
A13	78,1	7	Nenhuma interação	Desenvolia e acompanhava bem as atividades da aula, também com o uso do tutorial, quando não estava presente nos encontros. Atrasou a entrega de um App.
A14	90,6	7	Pouca interação.	Desenvolveu e acompanhou as atividades da aula, não apresentando maiores dificuldades. Entregou um App com atraso.
A15	64,6	7	Nenhuma interação	Desenvolia e acompanhava bem as atividades tanto em aula quanto aquelas a partir do tutorial. Disponibilizou dois App com atraso.
A16	84,4	7	Nenhuma interação	Desenvolia e acompanhava bem as atividades propostas em aula. Cumpriu com todos os App no tempo solicitado.
A17	55,2	7	Nenhuma interação	Desenvolia e acompanhava bem as atividades da aula, bem como através do tutorial. Disponibilizou dois App fora do prazo. Embora o sujeito se declare programador, os testes objetivos não evidenciaram essa informação. Conforme pode-se perceber no Projeto Final, sua nota ficou um pouco acima do mínimo, pois seu App não validou os dados de entrada, sendo a interface de difícil compreensão e consequentemente o teste não se torna uma tarefa fácil. Parte das perguntas do aplicativo estavam truncadas e quanto às respostas do Quiz, apenas duas estavam corretas.
A18	50,0	6	Nenhuma interação	Dificuldade de acompanhar as atividades da aula. Poucos App finalizados. Apresentou dificuldade nas operações e fórmulas matemáticas. Frequente solicitação para que a professora fosse verificar seu código. Interessante observação sobre este sujeito que foi o único que acertou a questão 2 do segundo teste, no qual as Habilidades 1, 3, 4 e 5 foram observadas.
A19	62,5	6	Pouca interação.	Desenvolia e acompanhava bem as atividades da aula e nas suas ausências fez uso do tutorial. Entregou dois App com atraso.

ID aluno	Avaliação Objetiva	App enviados/7	CoP	Observação
A20	79,5	4	Nenhuma interação	Bom programador, apresentou diferentes formas de programação em seu Projeto Final.
A21	46,9	6	Nenhuma interação	Muitas faltas, mas conseguiu acompanhar as atividades através dos tutoriais.

Fonte: Da autora (2020).

A partir da análise das informações apresentadas na Tabela 7, este estudo de caso resultou em 14 sujeitos como ACod, sendo que A05 e A19 iniciaram sua aprendizagem neste curso.

Julgou-se importante analisar mais profundamente três sujeitos, a saber, A06, A17 e A18, que declararam que sabiam programar, mas que após as avaliações não foi possível confirmar essa informação. O aprendente A06 respondeu a todos os testes, sendo que no segundo errou todas as questões. No pós-teste, das 5 questões propostas errou 3. Conforme já exposto, não compareceu a vários encontros e quando presente demonstrou dificuldade em acompanhar as atividades propostas. Ao observar suas respostas sobre se sabia programar, declarou que já havia participado de dois cursos de programação e que “*adora programar*”. Porém, ao ser indagado sobre “o que é um algoritmo?” respondeu que “*é um programa*”. Evidentemente que esta resposta está equivocada. Da mesma maneira, não foi capaz de expressar resposta à questão “defina comando de condição e de repetição”. O resumo das avaliações dos aplicativos desse sujeito está apontado na Tabela 8.

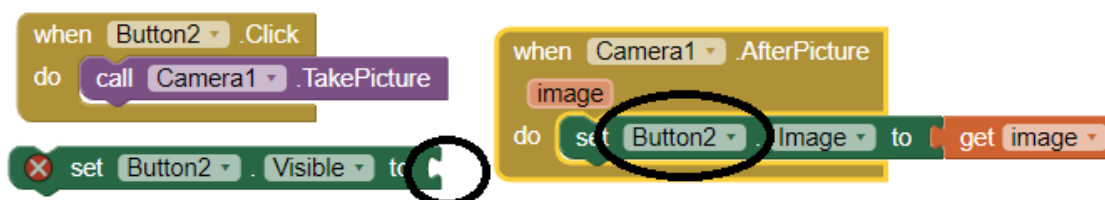
Tabela 8 - Resumo da avaliação dos App do sujeito A06

App	Descrição
Gato	Posicionou a foto no lugar errado.
Tradutor	Poucas alterações na interface, não completou a programação, ou seja, não fez 2 idiomas, conforme solicitado.
Combustível	Não concluiu o desenvolvimento do código em aula, e este apresentou erros de lógica.
Passos	Não enviou para avaliação.
Calculadora	Interface similar à do tutorial. Programação correta. Não implementou o botão de saída do App.
Fatorial	Interface similar à do tutorial. Programação correta. Não implementou o botão Limpar.
Cambio	Não enviou para avaliação.
Projeto Final	Variável Contador de acertos não foi programada corretamente.

Fonte: Da autora (2020).

No App Calculadora e Fatorial, o sujeito desenvolveu corretamente, porém, não completando todas as funcionalidades. A porção do código-fonte desenvolvido pelo sujeito, no App do Gato, é mostrada na Figura 44.

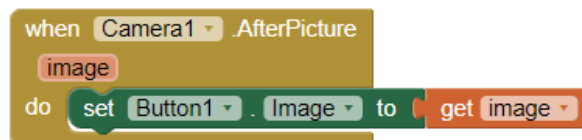
Figura 44 - Porção de código-fonte App Gato



Fonte: Da autora (2020).

Na Figura anterior, pode-se perceber que houve dois erros: o comando SET não foi finalizado e no comando WHEN, o *Button 2* está errado, deveria ser o *Button 1*, conforme Figura 45, o que demonstra a falta de compreensão e resolução de problemas.

Figura 45 – Parte do código-fonte correto do App Gato



Fonte: Da autora (2020).

Outro exemplo que se pode exibir é a programação do App Combustível. Na Figura 46 está mostrado o código-fonte desenvolvido pelo aprendiz.

Figura 46 - Porção código-fonte do App Combustível

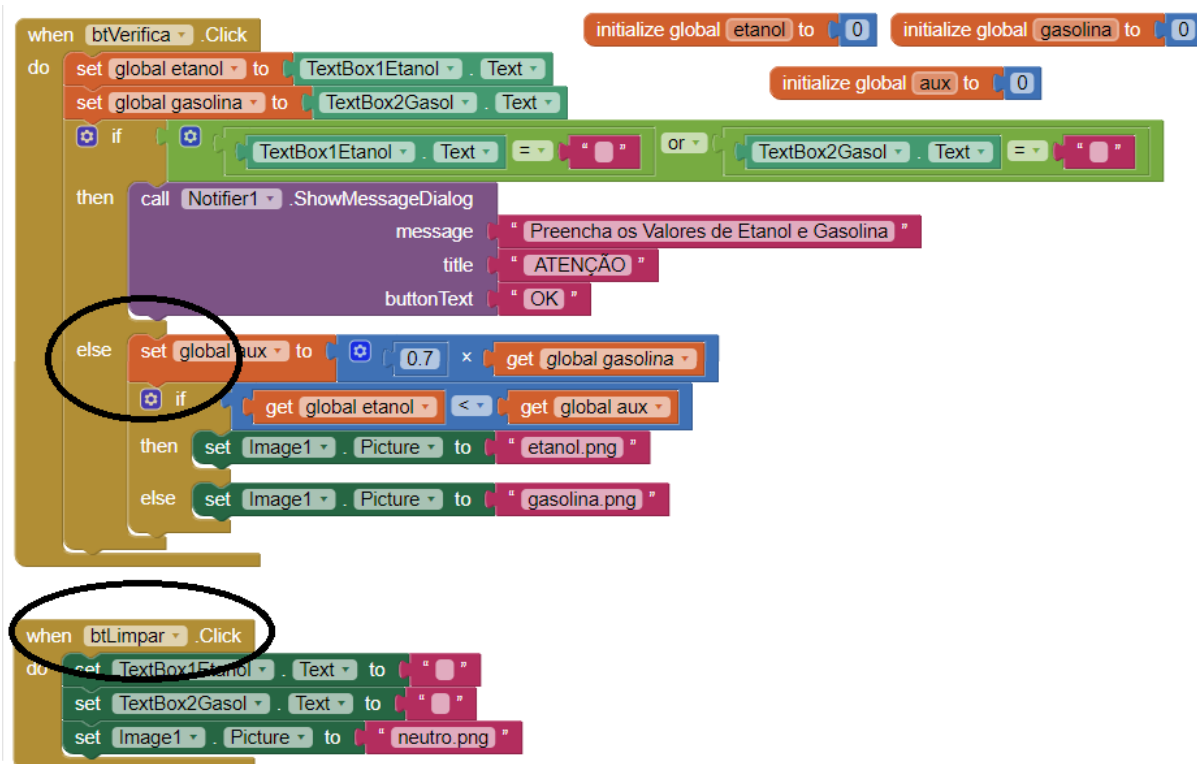
```

initialize global etanol to 0
initialize global gasolina to 0
initialize global aux to 0
when Button1 Click
do
  if
    TextBox1 . Text = "" or
  then
    call Notifier1 . ShowMessageDialog
      message "llene los valores..."
      title "ATENCION"
      buttonText "OK"
  else
    set global etanol to TextBox1 . Text
    set global gasolina to TextBox2 . Text
    set global aux to get global gasolina × 0.
    if
      get global etanol < get global aux
    then
      set Image1 . Picture to "etanol.png"
    else
      set Image1 . Picture to "gasolina.png"
  
```

Fonte: Da autora (2020).

Ao analisar a Figura 46, se observa que o comando IF, que está circulado, não se encontra conectado com a sequência de comandos iniciadas no comando *When*. Além disso, o bloco de comandos para limpar (formatar) os dados de entrada não constam no código, isso significa que ao App falta funcionalidade. Para um melhor entendimento, na Figura 47 se apresenta uma possível solução de programação para este App.

Figura 47 - Código-fonte sugerido para o App Combustível

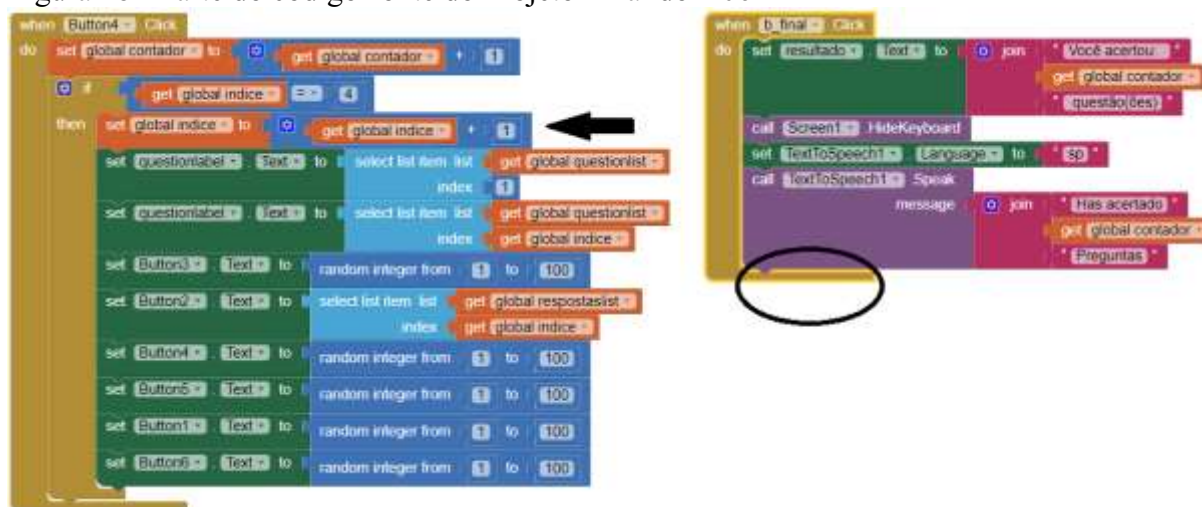


Fonte: Da autora (2020).

Os destaques em preto na Figura 47 apontam onde ocorreu a falha do sujeito na programação, sendo que no comando *Else/If* o erro foi de não completar a sequência lógica dos comandos. Já na segunda anotação foi a ausência deste bloco na implementação.

Nos aplicativos de Cambio e Fatorial do aluno A06, os códigos enviados estavam corretos, porém incompletos. Por fim, no código-fonte do Projeto Final demonstrou a falta de compreensão da linguagem de programação. Primeiramente, apresenta 6 botões que correspondem às possíveis respostas geradas randomicamente. Se o usuário selecionar o botão Sair a resposta apresentada está incorreta. No final do Quiz, como não foi verificado o final da LIST, o App mostra um erro (Figura 48). É perceptível que as habilidades compreensão (H1), resolução de problemas (H3) e resolução algorítmica (H4) não foram evidenciadas.

Figura 48 - Parte do código-fonte do Projeto Final do A06



Fonte: autora (2020).

Na Figura 48, o erro do final do aplicativo ocorreu porque a variável “índice” está sendo incrementada, porém faltou o teste de final da lista. Isso implica que, quando o ponteiro estiver direcionado para uma posição que extrapola o tamanho da lista de perguntas, o App gera um erro e é finalizado. Na parte da direita da mesma figura, se observa que a variável “contador” não está sendo zerada, isto implica um incremento contínuo, mesmo que o usuário retorne para o início do jogo.

A partir dessas análises nos códigos-fonte do A06, se confirma que ele não apresenta indícios suficientes para ser considerado ACod. Em síntese, após observado o resultado da avaliação objetiva e subjetiva, pode-se concluir que o sujeito pode ser considerado um pensador computacional desplugado (PCD).

O segundo aluno, A17, declarou que aprendeu a Linguagem C durante um semestre. Respondeu corretamente às indagações do questionário “Conhecendo Você” sobre o que são algoritmos, variáveis e comandos de condição e repetição. Não respondeu ao pós-teste, e dos demais acertou 63,6% (considerando somente os testes respondidos).

O resumo das avaliações dos App está descrito no Quadro 9.

Quadro 9 - Resumo da avaliação dos App do sujeito A17

App	Descrição
Gato	A programação não estava completa, pois não adiciona novas fotos. Adicionou vários sons e cores.
Tradutor	Não completou a programação, ou seja, não fez 2 idiomas, conforme solicitado. Adicionou vários sons e cores.

App	Descrição
Combustível	Não completou a programação, explicitando que faltaram os cálculos e as operações corretas, os quais são consideradas funcionalidades essenciais neste App. Entregou fora do prazo.
Passos	O código-fonte estava completo, porém, com erro semântico nos cálculos. Entregou atrasado.
Calculadora	Não completou a programação, faltando as funcionalidades de saída do App e o cálculo da potência 2 (exigida a implementação com o comando <i>While</i>). Interface semelhante a sugerida no tutorial.
Fatorial	Código-fonte incompleto, somente apresenta o cálculo realizado com o comando FOR. Entregou atrasado.
Cambio	A implementação do Botão Limpar não funcionou de forma correta, ou seja, a programação estava errada. O cálculo do troco foi implementado corretamente, porém a resposta na interface (saída) está na posição errada.
ProjFinal	Na interface, as descrições aparecem truncadas em duas questões, não sendo possível ler o enunciado completo. O resultado de ambas está apresentado de forma equivocada, ou seja, a resposta que o sujeito calculou e programou está errada.

Fonte: Da autora (2020).

A partir do exposto no Quadro 9, se pode concluir que o sujeito não completou nenhum aplicativo, além de alguns apresentarem erros nas operações matemáticas. Para melhor elucidar as observações realizadas, a seguir apresentam-se alguns exemplos. Na Figura 49, que mostra a interface do aplicativo Passos, observa-se que o usuário selecionou a opção lenta para caminhar, executa 21 passos e gasta 3033,97 Kcal (quilocaloria), o que não condiz com a realidade e com a fórmula indicada no tutorial. O resultado correto para as Kcal gastas é de 10,44.

Figura 49 - App Passos do A17



Fonte: Da autora (2020).

A saída da variável Kcal mostrada na Figura 49 vem a partir do erro na programação do código-fonte (Figura 50). Ademais, pode-se perceber na interface que não está agradável para o usuário, contendo símbolos de uma calculadora e de moeda no segundo plano, tornando-a visualmente poluída.

Figura 50 - Parte do código-fonte do Projeto Final do A17

```

when Pedometer1 .SimpleStep
  simpleSteps distance
do
  set global passo to get simpleSteps
  set TextBoxPAssos .Text to join " Passos: " get global passo
  set TextBox2 .Text to join " Distância: " get distance
  if get global passo = 3000
  then
    call Sound1 .Play
    call Sound1 .Vibrate
      milliseconds 500
    set TextBox3 .Text to join " 3000 passos..." Pedometer1
  if get global peso ≠ 0 and get global peso
  then
    set global tipopasso to get global peso *
    set TextBox3 .Text to join " KCAL: " get global kcal
  else
    set TextBox3 .Text to " KCAL: valor nulo "

```

Fonte: Da autora (2020).

Na Figura anterior, é possível verificar que o erro do programador foi o de atribuir todo o cálculo das Kcal para a variável “tipopasso”, ao invés da variável “kcal”, ambas indicadas na Figura. Ou seja, o cálculo correto está armazenado em “tipopasso”, porém na saída é enviado o valor da variável “Kcal”. Outro exemplo da falta de completude dos códigos-fonte é apresentado na Figura 51, relativo ao App Fatorial.

Figura 51 - Código-fonte do App Fatorial de A17

```

when BotónPARA .Click
  initialize global numero to 0
  initialize global resultado to 1
do
  set global resultado to 1
  for each number from 1
    to get global numero
    by 1
  do
    set global resultado to get global resultado * get global numero
    set global numero to get global numero - 1
  set CampoDeTextoresultado .Text to get global resultado

when BotónMientras .Click
do
  get global numero

```

Fonte: Da autora (2020).

A Figura 51 apresenta o código-fonte enviado pelo aluno e os dois comandos requeridos: *FOR* e *WHILE*. Este último não foi sequer iniciado. O botão de Limpar não foi codificado e, mais importante, o valor de entrada não foi utilizado no cálculo porque não foi lido, resultando sempre o valor 1 (o qual é atribuído no início da codificação para a variável “resultado”). A Figura 52 mostra um exemplo da interface com o erro.

Figura 52 - Interface do App Fatorial de A17



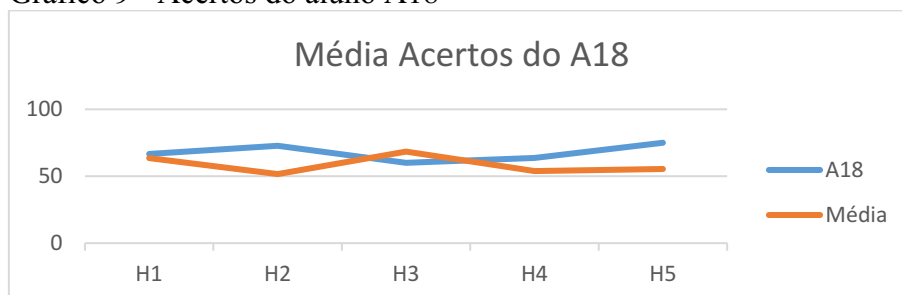
Fonte: Da autora (2020).

Na Figura anterior que apresenta o App, na entrada do valor 5 resulta em seu fatorial igual a 1, onde deveria aparecer o valor de 120, pois conforme já comentado anteriormente o valor de entrada não é considerado no cálculo.

Portanto, pode-se concluir que o aprendente A17 não pode ser considerado ACod, e sim um pensador computacional desplugado (PCD), pois está bastante próximo dos 60% sugeridos no MRPC, porém ainda não conseguiu se apropriar dos comandos necessários.

E, por fim, tem-se o sujeito A18, o qual manifestou que fez curso superior na área de Computação, porém há muito tempo, mas que nunca trabalhou na área de programação. Ainda declarou o conhecimento em 4 linguagens de programação. Respondeu a todos os testes propostos, sendo a quarta melhor nota da turma. O Gráfico 9 evidencia o desempenho deste sujeito em número de acertos por habilidades com relação à média da turma.

Gráfico 9 - Acertos do aluno A18



Fonte: Da autora (2020).

O referido sujeito, somente na Habilidade 3, confirmado no Gráfico anterior, ficou com sua média inferior em relação aos demais sujeitos da pesquisa. O resumo das avaliações dos App enviados está no Quadro 10.

Quadro 10 - Resumo da avaliação dos Apps do A18

App	Descrição
Gato	Desenvolveu seu App igual ao tutorial.
Tradutor	Faltou colocar a saída da tradução em espanhol no local correspondente. Interface igual ao tutorial. Erro de lógica na chamada ao segundo tradutor.
Combustível	Desenvolveu seu App igual ao tutorial.
Passos	Não disponibilizou o aplicativo.
Calculadora	Desenvolveu seu App igual ao tutorial.
Fatorial	Desenvolveu seu App igual ao tutorial.
Cambio	Desenvolveu seu App igual ao tutorial.
Projeto Final	Não foi possível avaliar o funcionamento do App, o qual já na primeira questão apresentada ocorreu um erro e em seguida é finalizado.

Fonte: Da autora (2020).

A partir da análise das informações descritas no Quadro 10, pode-se somente analisar o código-fonte do App Tradutor (Figura 53), visto que nos demais os códigos-fonte seguem o tutorial, não sendo agregados nenhum comando ou funcionalidade aos aplicativos.

Figura 53 - Código-fonte do App Tradutor do A18

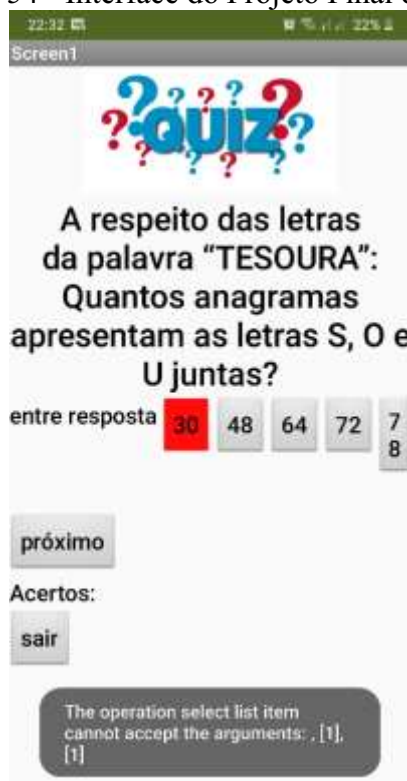
Código-fonte do aluno	Código-fonte correto

Fonte: Da autora (2020).

Na análise dos códigos-fonte apresentados na Figura 53, pode-se concluir que o aluno não ancorou de forma correta o uso dos comandos e sua sequência lógica, pois a chamada ao componente Tradutor (*YandexTranslate*) deve ser colocada quando o botão de tradução for acionado (*b_traduzir*), e não quando já havia sido realizada a tradução.

Já para o Projeto Final, o código-fonte não será apresentado, pois é bem extenso, podendo ser acessado em: <http://ecompartindo.santana.ifsul.edu.br/mod/forum/discuss.php?id=736>. A interface é mostrada na Figura 54.

Figura 54 - Interface do Projeto Final do A18



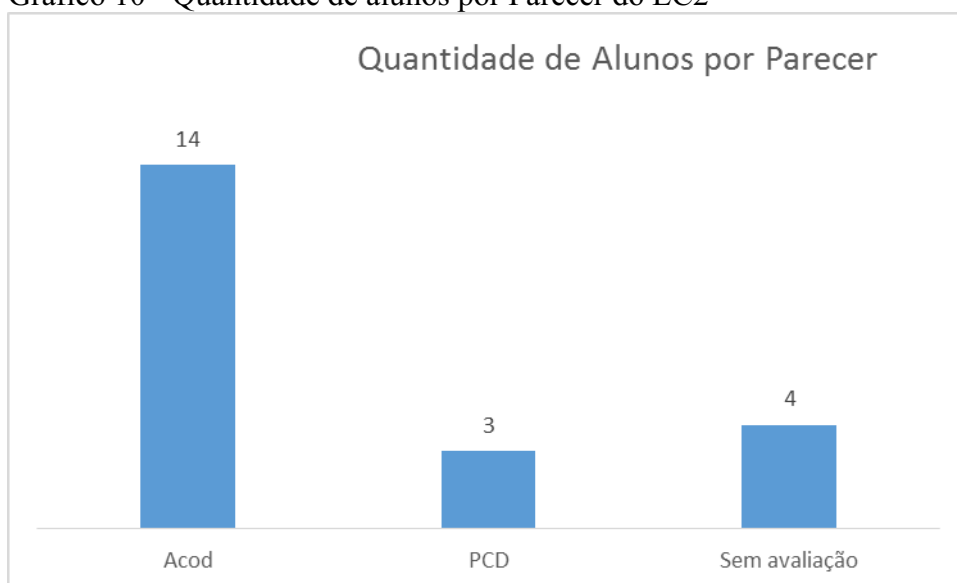
Fonte: Da autora (2020).

Ao iniciar, o App mostra a primeira pergunta, que no caso a resposta escolhida foi 30 (está destacada na cor vermelha), independente da resposta (toque no botão “próximo”), a janela de erro é exibida (Figura 54). Portanto, não foi possível realizar os testes necessários. Ao ser preenchido o documento Avaliação do Projeto Final (Figura 27), esse aplicativo somente obteve pontuação na subcaracterística funcionalidade e operabilidade.

Em resumo, estes sujeitos analisados (A06, A17 e A18), embora afirmaram que possuíam o conhecimento de programação antes de iniciar o curso “Eu Programa 1.0!”, não foi possível reconhecê-los como ACod. Porém, pode-se reconhecer que são sujeitos PCD, ou seja, pensadores computacionais desplugados: os sujeitos A06 e A18, pelo seu bom desempenho nos testes, e A17, pelos indícios de aprendizagem significativa ao longo do processo, considerando seu desempenho nos testes e no desenvolvimento do Projeto Final.

O resumo da classificação dos sujeitos, após a avaliação, pode ser visualizado no Gráfico 10.

Gráfico 10 - Quantidade de alunos por Parecer do EC2



Fonte: Da autora (2020).

Nessa representação se pode afirmar que 66,66% dos alunos são considerados ACod. Porém, foram 15 sujeitos que no início assumiram que sabiam programar. Deste conjunto, 3 foram considerados PCD e 2 alunos que não sabiam programar se apropriaram de conhecimentos e foram considerados ACod.

Retomando o MRPC, um sujeito ACod pode ser considerado LCod quando apresenta o conhecimento de mais de uma LP e comandos que não estão listados no modelo. Isso posto, se investigou os sujeitos que declararam que sabiam programar no início do curso e que mantiveram esta classificação após o término dos encontros. A análise, por aluno, está descrita no Quadro 11.

Quadro 11 - Alunos com status LCod

ID aluno	Parecer	Análise	Parecer para LCod
A01	ACod	Declarou que tem experiência em 3 LP, porém não codificou nenhum comando diferente. Portanto não pode ser considerado Lcod.	
A02	ACod	Declarou que tem experiência em 2 LP. No código do projeto final implementou chamadas de funções e várias interfaces.	LCod
A03	ACod	Declarou que conhece 4 LP. Fez uso de chamada de procedimentos e várias interfaces no projeto final.	LCod
A04	ACod	Declarou que fez o ensino médio integrado com o Curso de Informática e codifica em três LP. Não foram encontrados comandos diferentes nos códigos-fonte apresentados.	

ID aluno	Parecer	Análise	Parecer para LCod
		O seu projeto final não estava completo, nem correto; não apresentou validação dos dados de entrada e a interface não estava amigável ao usuário.	
A05	ACod	Declarou que não conhecia programação. Portanto não poderá ser LCod, sendo está sua primeira experiência em programação.	
A08	ACod	Declarou que conhece 3 LP. Implementou chamada de procedimentos e tratamento de interface com <i>scroll</i> .	LCod
A10	ACod	Declarou que conhece 2 LP. Não foram encontrados comandos diferentes nos códigos-fonte apresentados.	
A12	ACod	Declarou que fez o curso técnico Informática para Internet, portanto já sabia programar. Incrementou nos seus códigos-fonte passagem de parâmetro e chamadas de várias interfaces.	LCod
A13	ACod	Declarou que possui um curso técnico, porém incompleto. Não foram encontrados comandos diferentes nos códigos-fonte apresentados, portanto não pode ser considerado LCod.	
A14	ACod	Afirmou que é programador em quatro LP. Apresentou código-fonte com chamadas a procedimentos.	LCod
A15	ACod	O sujeito se declarou programador Web e conhece 3 LP. Apresentou no código-fonte do projeto final a chamada de interface e validações de dados de entrada.	LCod
A16	ACod	Declarou que fazem 3 anos que está na área de programação e que cursou técnico de Informática para Internet. Apresentou no seu código-fonte o uso de várias interfaces e o comando de Lista de Lista.	LCod
A19	ACod	Declarou que não conhecia programação. Portanto não poderá ser LCod.	
A20	ACod	Declarou que fez o curso técnico Informática para Internet. No seus códigos-fonte implementou passagem de parâmetros e várias interfaces. Já no Projeto Final fez uso de uma variável de retorno para a interface principal.	LCod

Fonte: Da autora (2020).

Dos 14 sujeitos classificados como ACod, 8 foram considerados como LCod (Quadro 11). Destes, 5 são homens e uruguaios, os demais são brasileiros e dentre eles um sujeito do sexo feminino. A média dos sujeitos letrados em código subiu para 81,75 se comparados a 68,29 de toda a amostra do EC2, e somente o aluno A15 ficou com sua nota abaixo da média da turma. Este resultado retrata a condição dos alunos, ditos programadores, ou, em outras palavras, alfabetizados em código, que ancoraram os conhecimentos de uma nova LP (pois nenhum dos sujeitos da amostra declarou o conhecimento com o *App Inventor*). Com respeito à aquisição de uma nova LP nestes sujeitos, presumivelmente, ocorreu uma modificação nos subsunçores, como bem explicam Ausubel, Novak e Hanesian (1983):

[...] é importante reconhecer que aprendemos um novo código sintático empregando como modelo a sintaxe da própria linguagem e depois observando as semelhanças e as diferenças entre ambas. [...]. Há vários aspectos do conhecimento da primeira

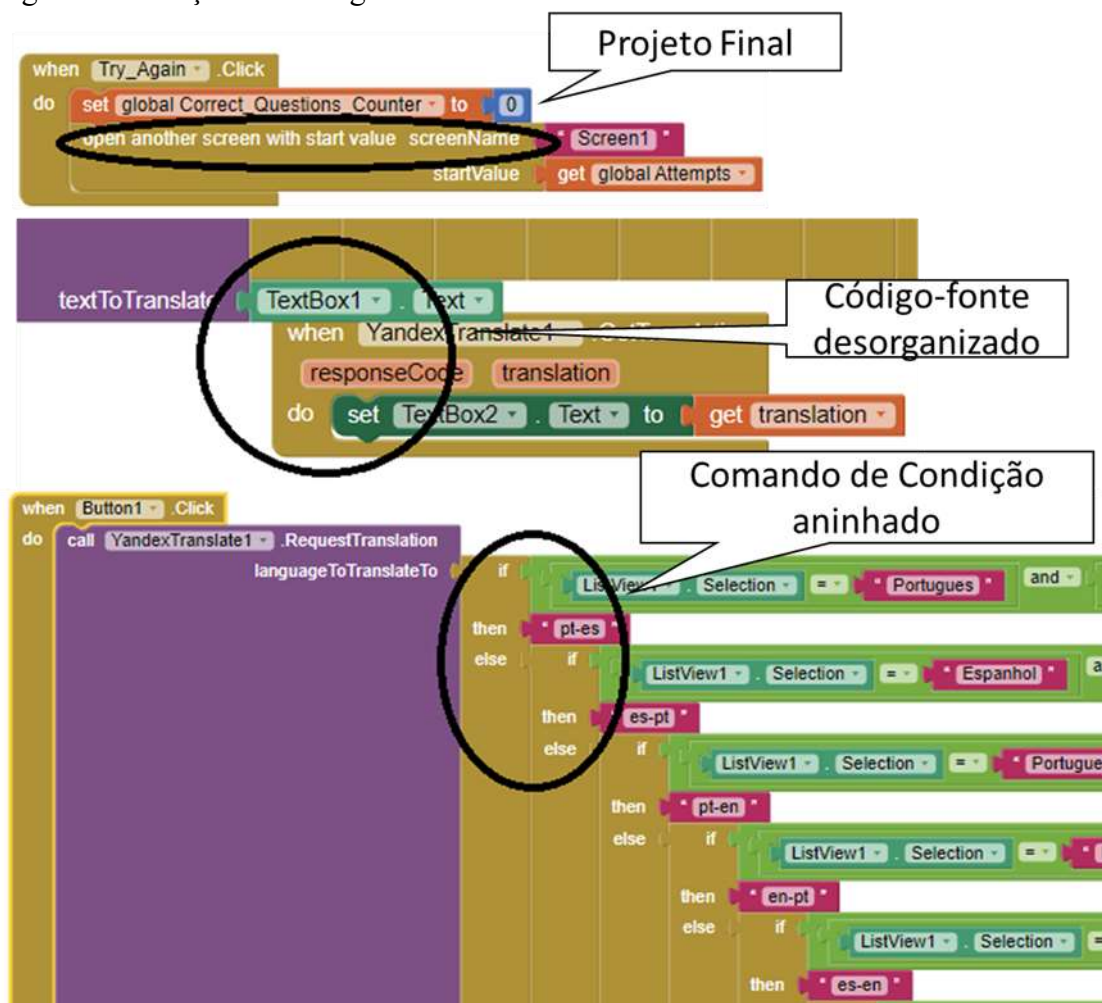
linguagem: o significado da maioria dos conceitos, a compreensão das categorias e as funções sintáticas, a facilidade de empregar padrões estruturais que podem ser quase idênticos em ambas linguagens e que são transferíveis diretamente para a aprendizagem da segunda linguagem (AUSUBEL; NOVAK; HANESIAN, 1983, p. 78, tradução nossa).

Para os sujeitos restantes, aqueles declarados ACod, porém não considerados LCod, Perrenoud (1999b, p. 17) explica através do entendimento das competências:

- Se a competência se manifesta na ação, não é inventada na hora:
- se faltam os recursos a mobilizar, não há competência;
 - se os recursos estão presentes, mas não são mobilizados em tempo útil e conscientemente, então, na prática, é como se eles não existissem.

A partir das observações anotadas no Quadro 11, se apresentam alguns trechos de código-fonte para validar o parecer final sobre letramento em código. Na Figura 55, a evidência do sujeito A02.

Figura 55 - Porções do código-fonte do A02

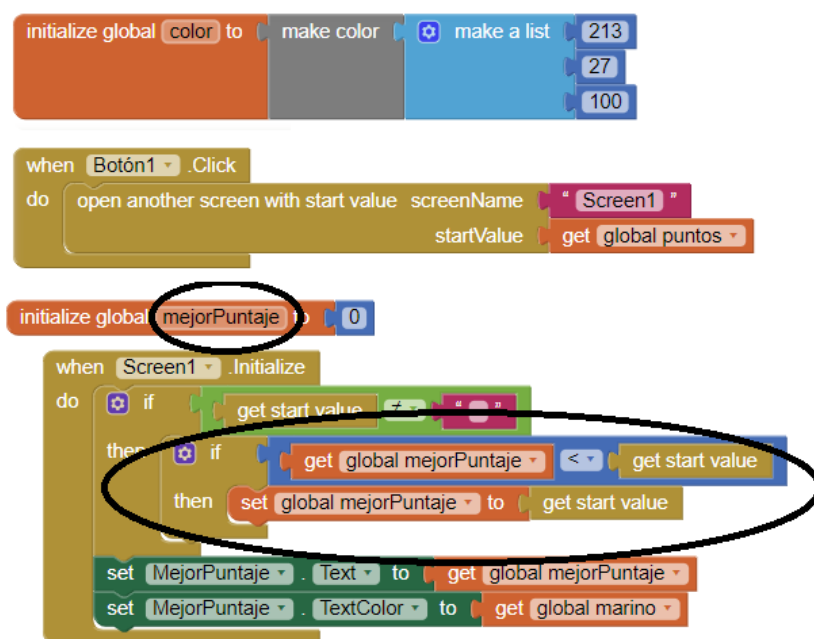


Fonte: Da autora (2020).

O sujeito A02, considerado LCod, embora seu código esteja desorganizado (identificado na Figura 55), apresentou o comando de condição (*if/then/else*) de forma aninhada e a abertura de uma nova janela com passagem de parâmetros, ambos considerados novos comandos com relação ao estabelecido no MRPC.

Na Figura 56 tem-se trechos do código-fonte do aluno A20.

Figura 56 - Fragmentos de código-fonte do A20



Fonte: Da autora (2020).

A partir da análise da Figura 56, esse aluno implementou a programação de uma variável para armazenar e mostrar o maior número de acertos, utilizando três interfaces (bloco intermediário) e no primeiro bloco faz a programação de cores através de uma Lista. Nenhum desses componentes foi apresentado em aula. Julgou-se interessante apresentar as interfaces do App ao usuário (Figura 57).

Figura 57 - *Feedback* do Projeto Final do A20



Fonte: Da autora (2020).

Na Figura anterior está mostrada a interface de *feedback* que o App apresenta ao usuário, dependendo se a resposta para a questão foi correta ou não. Essas interfaces exigiram um conhecimento que o aluno buscou através de aprendizagem por descoberta, pois não foram mencionados durante os encontros.

No final do curso foi oferecido um espaço para os sujeitos descreverem suas sugestões e críticas no AVA. No final do Caso Piloto, ou EC1, não houve contribuições e/ou críticas com relação ao curso. Neste EC, a liberação do certificado de conclusão do curso ficou vinculado a sua opinião final. Foram 14 sujeitos que enviaram seus pareceres, sendo a maioria avaliando de forma positiva, segundo pode ser acompanhado em algumas citações:

Conseguimos aprender a programação como um todo, cada aplicativo feito foi diferente um do outro, abrangendo vários cálculos distintos, acredito que vamos usar muito esse programa novo e fazer vários apps diferentes e divertidos. Obrigada, profe!

Ótimo curso, com uma curva de aprendizagem muito bem trabalhada, mesmo para alunos com pouco conhecimento na área de informática, tornando assim um curso acessível a todos.

Todo el curso estuvo excelente solamente una sugerencia para la próxima edición se podría profundizar el uso de bases de datos para el almacenamiento y procesamiento de información.

¡Muy buena e interesante enseñanza!, aprendí a hacer aplicaciones de celulares de un modo sencillo e incluso divertido.

Em síntese, este estudo de caso conseguiu aplicar e validar o MRPC de forma completa, pois os sujeitos demonstraram a ancoragem de habilidades para serem considerados ACod, e dentre este grupo alguns obtiverem o *status* de LCod, conforme pode ser legitimado nos

códigos-fonte dos aplicativos enviados e principalmente no Projeto Final. Na sequência, estão narradas algumas reflexões sobre a diferença dos testes realizados no EC1 e no EC2.

6.2 Considerações sobre a quantidade de testes

Neste estudo de caso, conforme já anotado, foram aplicados 4 testes e no EC1 3 testes, devido à diferença de conhecimentos declarados pelos sujeitos sobre programação. Apresenta-se nesta seção uma análise sobre os resultados de ambos os testes.

No Gráfico 11, está demonstrada a diferença das médias do EC2, considerando a realização de 3 testes e de 4 testes para os alunos A01 ao A10. Já no Gráfico 12, os dados são dos alunos A11 ao A21. Os dados foram divididos em duas partes para efeito de uma melhor visualização.

Gráfico 11 - Diferença das Médias – Parte 1

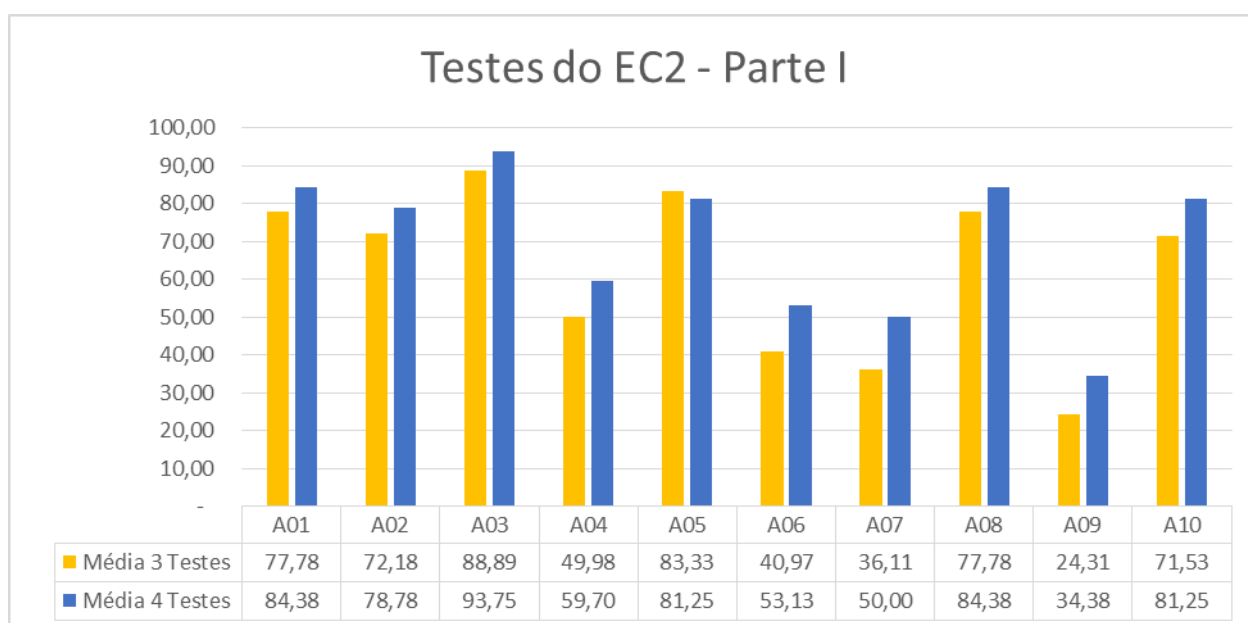
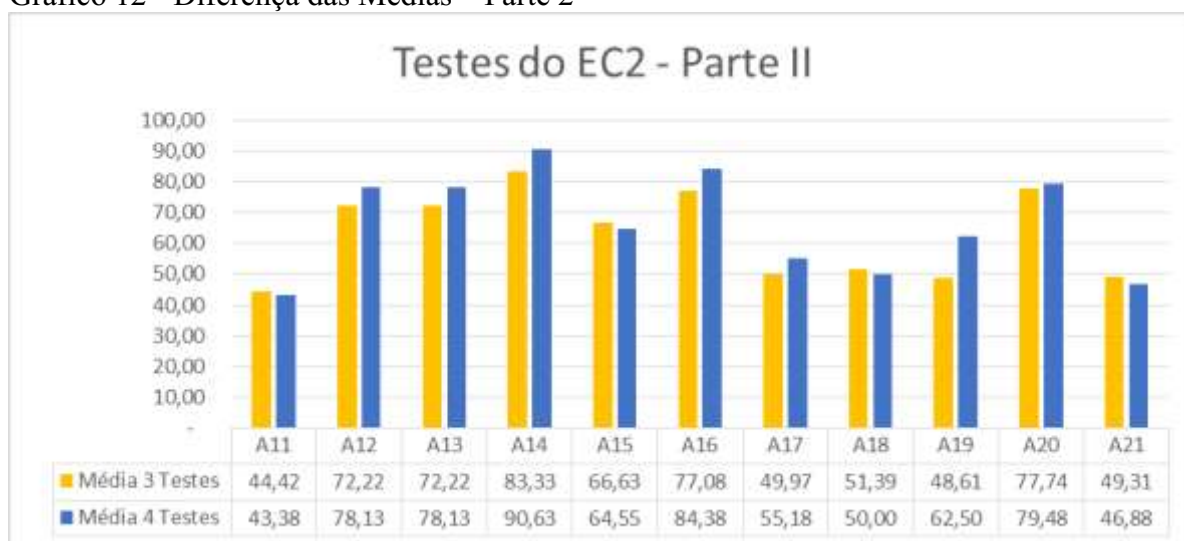


Gráfico 12 - Diferença das Médias – Parte 2



Fonte: Da autora (2020).

Para apresentar a variação das médias dos sujeitos entre os dois testes, a Tabela 9 foi definida.

Tabela 9 - Variação das Médias dos Testes no EC2

ID Aluno	Variação Percentual	ID Aluno	Variação Percentual	ID Aluno	Variação Percentual
A01	8,5%	A08	8,5%	A15	-3,1%
A02	9,1%	A09	41,4%	A16	9,5%
A03	5,5%	A10	13,6%	A17	10,4%
A04	19,5%	A11	-2,3%	A18	-2,7%
A05	-2,5%	A12	8,2%	A19	28,6%
A06	29,7%	A13	8,2%	A20	2,2%
A07	38,5%	A14	8,7%	A21	-4,9%

Fonte: Da autora (2020).

A partir dos dados apontados nos Gráficos e na Tabela anteriores, somente A04 e A19 teriam sua classificação modificada de ACod para PCD, para ambos, segundo o MRPC. Observando-se o valor do A19: ao responder três testes foi de 22,22% e com 4 testes foi de 57%, sem considerar o valor do Projeto Final. No primeiro caso (3 testes), o sujeito seria considerado PCD ou não atingiu, e, no segundo caso, sua classificação aponta para ACod. Percebeu-se um aumento no valor de sua nota, acompanhando a tendência da amostra.

Já o sujeito A04, ao responder somente 3 testes obteve um valor de, aproximadamente, 55,5, e para 4 testes sua nota elevou-se para 75. Observando o valor final da média com 4 testes

– 59,7 – foi arredondada para 60, pois o sujeito declarou que conhece várias LP (linguagem de programação). Percebeu-se um aumento no valor de sua nota, acompanhando a tendência da amostra em A04 e A19.

Em ambos os casos somente houve um reposicionamento dos sujeitos perante a classificação do MRPC, ainda que, com essas observações destacadas, pode-se inferir que a diferença entre aplicar três ou quatro testes pouco modifica a classificação final, embora nos casos apresentados os dois alunos teriam sua classificação apontando para ACod, fundamentada nos três componentes da avaliação subjetiva.

Além disso, apenas cinco sujeitos tiveram uma variação percentual negativa, isto é, suas médias ficaram menores ao responder 4 testes, porém com valores inferiores a 5% (Tabela 9).

Portanto, o uso do MRPC pode ser adaptado com relação ao número de testes, conforme o grupo de sujeitos em questão, sempre considerando as necessidades ou a demanda dos aprendizes e sua aprendizagem significativa. Assim, cabe ao tutor/professor a decisão e/ou necessidade de desenvolver novos testes. Isso posto, pode-se afirmar a eficiência da aplicação do MRPC, destacando-se que para esta amostra 80% dos sujeitos não tiveram sua classificação alterada.

Ao finalizar este capítulo, mais precisamente os resultados encontrados nos dois estudos de caso, é importante ressaltar as respostas confirmadas para as hipóteses **a**, **b** e **c** definidas nesta tese.

No capítulo seguinte, as conclusões e os apontamentos quantitativos da pesquisa estão percorridos, além do resultado da última hipótese.

7 RESULTADOS FINAIS E DISCUSSÕES

No Capítulo 5 foi apresentado o Caso Piloto ou Estudo de Caso 1 (EC1), no seguinte, o EC2 foi detalhado. Neste capítulo, são apresentados os dados quantitativos sobre cada um dos estudos de caso, bem como a combinação de ambos os resultados referentes à diferença de nacionalidade, e, por fim, a análise das correlações entre as habilidades propostas no Modelo de Referência para o Pensamento Computacional (MRPC).

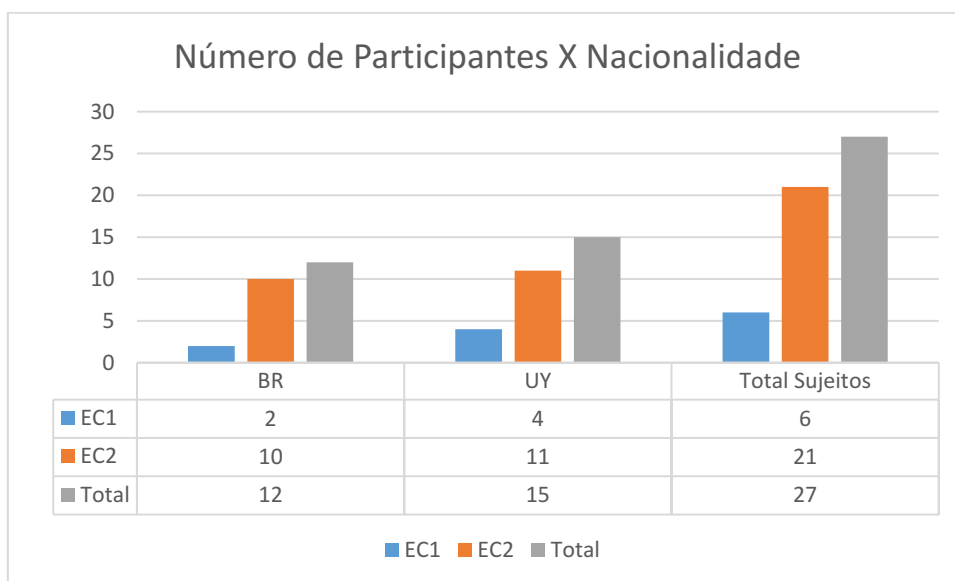
Ao analisar os dados obtidos da Avaliação Objetiva, segundo o MRPC, que é a média entre o valor dos testes realizados com o valor obtido na avaliação do Projeto Final, algumas considerações precisam ser elencadas e estão listadas a seguir.

Considerando que no EC2 foram realizados 4 testes, para fins de comparação entre EC1 e EC2 somente foram consideradas as questões semelhantes nos dois estudos. Isso quer dizer que os valores considerados nos cálculos correspondem a somente 3 testes em ambos os estudos de caso. Todos os dados do EC2 foram recalculados a fim de serem comparados com o EC1.

Cabe esclarecer que para as análises apresentadas no desenrolar deste capítulo, quando os sujeitos são classificados segundo sua nacionalidade, com a intenção de padronizar a nomenclatura e facilitar a compreensão dos dados os caracteres BR significam o conjunto de sujeitos com nacionalidade brasileira e UY, para os nascidos no Uruguai.

O resumo sobre a quantidade de sujeitos envolvidos nesta pesquisa, organizados por estudo de caso, é mostrado no Gráfico 13.

Gráfico 13 - Sujeitos X Nacionalidade

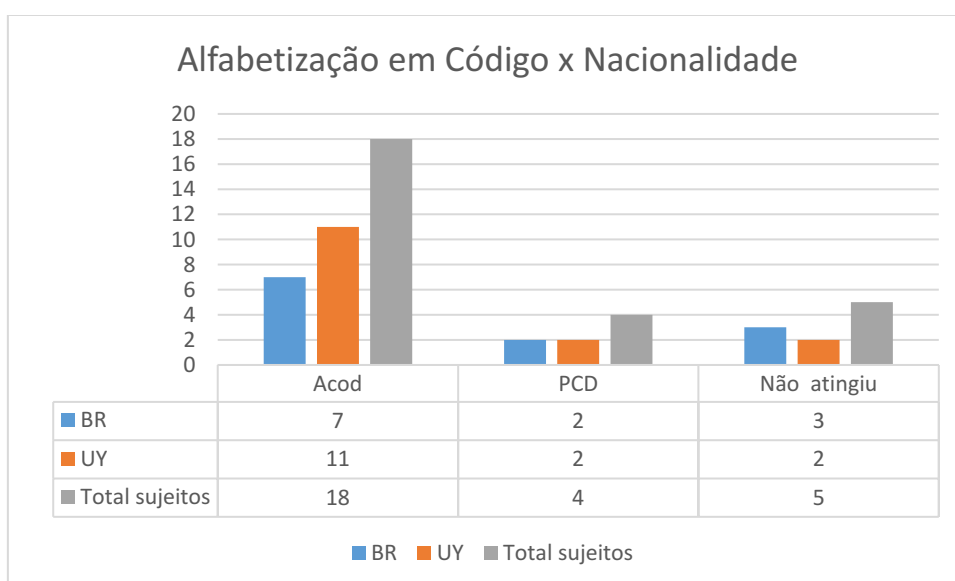


Fonte: Da autora (2020).

A partir do exposto, pode-se inferir que o total de sujeitos envolvidos (entende-se como a soma dos participantes do EC1 e do EC2) foram 27 alunos, sendo 12 brasileiros e 15 (ou seja, 55,55%) de nacionalidade uruguaia. Além disso, no primeiro estudo, trabalhou-se com 6 sujeitos e no segundo foram 21 alunos envolvidos.

Já no Gráfico 14 pode-se perceber a distribuição dos sujeitos catalogados por sua nacionalidade e classificação quanto ao desenvolvimento do PC.

Gráfico 14 - Alfabetização x Nacionalidade



Fonte: Da autora (2020).

Infere-se, a partir do Gráfico anterior, que o número de sujeitos classificados como pensador computacional desplugado (PCD) são iguais em ambas as nacionalidades.

Do total da amostra estudada, 11,1% são brasileiros e 7,4% são uruguaios. Ou seja, para 18,5% do conjunto de sujeitos não foi possível encontrar evidências de desenvolvimento de habilidades do PC.

Ao verificar-se a coluna dos sujeitos ACod (Gráfico 14), do total dos estudos de caso (N=27), obteve-se que 66,66% dos sujeitos avaliados foram classificados como alfabetizados em código, destes 25,9% são brasileiros e os demais são uruguaios (40,7%). Seguindo esta classificação, dos 12 sujeitos brasileiros envolvidos nos estudos de caso, 58,33% foram classificados como ACod, enquanto que para os alunos uruguaios o valor sobe para 73,33% (no caso, 15).

Portanto, os índices dos alunos de nacionalidade uruguaia (que também quer dizer de formação acadêmica no seu país de origem) são mais altos se comparados aos alunos brasileiros. Porém, para confirmar esta conclusão, se faz necessário analisar os dados de outra maneira:

Ao analisar os dados de uma amostra, devemos estar atentos ao fato de que algumas diferenças podem ser meramente casuais, ocasionadas por características próprias da amostra, não representando, necessariamente, propriedades da população que gostaríamos de conhecer. Neste contexto, torna-se importante estudarmos os chamados modelos probabilísticos [...], que constituem uma forma de mensurar a incerteza e, em consequência, fornecem uma metodologia adequada para generalizar resultados da amostra para a população (BARBETTA, 2002, p. 16).

Apropriando-se dos conceitos abordados na Estatística Descritiva, na Tabela 10, podem ser visualizados os valores os quais representam os resultados obtidos a partir das variáveis contínuas (STEVENSON, 2001) dos estudos de casos, ou seja, da amostra da população observada.

Tabela 10 - Resumo dos dados descritivos de EC1 e EC2

	Média Final	Desvio Padrão (DP)	Coefficiente de Variação - (CV)	Min	Max	Mediana	Moda
EC1	59,43	13,88	23%	38,67	72,33	66,78	66,78
EC2	62,66	18,18	29%	24,31	88,89	71,53	77,78

	Média Final	Desvio Padrão (DP)	Coefficiente de Variação - (CV)	Min	Max	Mediana	Moda
EC1+EC2	61,9	17,12	28%	24,31	88,89	66,78	66,78

Fonte: Da autora (2020).

A partir do exame dos dados da Tabela anterior, pode-se inferir que o valor final da avaliação objetiva (coluna Média Final), ficou acima dos 60% mínimos sugeridos utilizando o modelo proposto. Ou seja, a proposta do MRPC e sua aplicação no EC1 e EC2 pode ser considerada correta. Portanto, responde-se afirmativamente à hipótese **a**: o sujeito pode desenvolver as habilidades de PC e ser classificado como ACod.

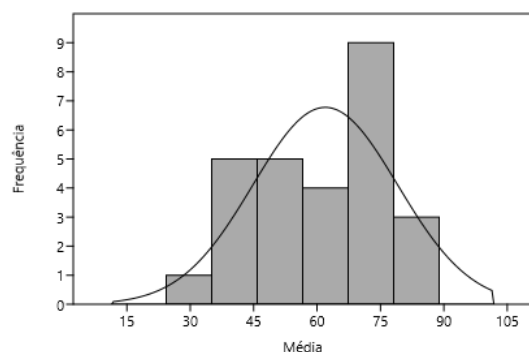
Quanto às medidas de dispersão, o desvio padrão (DP) e o coeficiente de variação (CV) indicam uma distribuição de dados não homogênea, classificada como regular com relação à média. Ainda apontam que a amostra de EC1 foi menos dispersa com referência ao segundo estudo de caso. O CV é entendido como a variabilidade dos dados em relação à média (SHIMAKURA, 2005), ou, em outras palavras, “indica o qual grande é o desvio padrão em relação à média” (ANDERSON *et al.*, 2019, p. 104). É consenso entre os autores que não existe uma tabela ou faixas que orientem a avaliação do CV (GARCIA, 1989; MOHALLEM *et al.*, 2008; SHIMAKURA, 2005); portanto, nesta tese, os valores inferiores a 15% são considerados baixos, quer dizer, razoavelmente homogêneos; entre 15% e 30% o CV é classificado como médio ou regular, e acima de 30% o CV é alto.

O valor que aparece na coluna Mediana significa que ele separa os valores em dois subconjuntos de mesmo tamanho para um conjunto ordenado de dados, e como se pode observar, estes valores são sempre maiores que 60. Então, pode-se concluir que o constructo andragógico empregado com os sujeitos obteve um bom resultado, pois mais da metade dos alunos atingiram o valor mínimo de 60%. Já a Moda, última coluna da Tabela 10, representa o valor que mais aparece, quer dizer, que mais se repete no conjunto, sendo que seu valor final está igual à Mediana. Portanto, a mesma conclusão sobre o uso adequado do constructo andragógico é novamente corroborada.

Na coluna Min está demonstrada a menor nota encontrada; assim tem-se que um sujeito do EC2 obteve o valor de 24,31, e outro, deste mesmo grupo, o valor máximo de 88,89 demonstrado na coluna Max.

O Gráfico 15 ilustra os dados das médias de todos os sujeitos envolvidos na pesquisa.

Gráfico 15 - Histograma da Média de todos sujeitos



Quadro 12- Dados da Amostra

Amostra Total

Média	61,938
Mediana	66,778
Moda	77,778
Desvio Padrão	17,119
Variância da Amostra	293,055
Coefficiente de assimetria	-0,424
Mínimo	24,306
Máximo	88,889
Nível de confiança (95,0%)	6,772

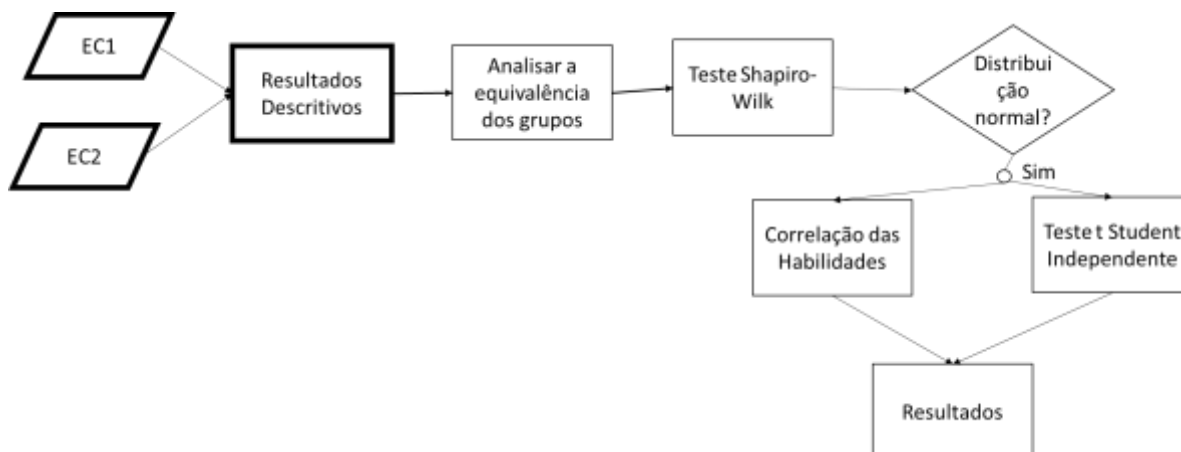
Fonte: Da autora (2020).

No histograma das médias, apresentado no Gráfico 15, pode-se inferir que a distribuição dos dados ocorreu de forma assimétrica à esquerda⁷², corroborado com o coeficiente de assimetria de -0,424 (Quadro 12). Isso quer dizer que a maior concentração de dados está nos valores mais altos (direita). Além disso, a Moda (valor que mais se repete no conjunto) é maior que a Mediana que, por sua vez, é maior que o valor da média. Com base nesses valores, reforça-se a resposta afirmativa para a hipótese “a” da tese.

A Figura 58 ilustra um fluxograma para facilitar a compreensão dos passos realizados com as duas amostras de dados, sendo que os que estão destacados foram descritos anteriormente.

⁷² O coeficiente de assimetria negativo (especialmente inferior a -1) indica que a cauda do gráfico é mais longa para a esquerda (BARBETTA, 2002).

Figura 58 - Fluxograma dos testes



Fonte: Da autora (2020).

Com o intuito de seguir analisando os dados, contou-se com o apoio dos *softwares* livres e gratuitos Past⁷³ e PSPP⁷⁴. O primeiro passo para avaliar os dados foi o estudo e análise a fim de determinar se os dados obtidos nos Estudos de Caso atendem a uma distribuição normal, pois este resultado determina os tipos de testes estatísticos que podem ser aplicados, complementando-se que existe uma grande quantidade desses testes (paramétricos e não paramétricos) e há de se considerar que:

[...] é a implicação prática de uma *diferença significativa estatisticamente*. Uma diferença significativa é uma diferença que não deve ter ocorrido meramente por acaso, mas não, necessariamente, é uma diferença relevante em termos práticos. [...]. Resta a análise prática para verificar se estas diferenças, que podem ser estimadas pelos dados, são relevantes (BARBETTA, 2002, p. 240).

Para os testes paramétricos é exigido que as amostras dos dados tenham uma distribuição normal, principalmente se tiverem uma dimensão (total da amostra) inferior a 30 (MORAES, 2016). Utilizou-se, nesta tese, o teste *Shapiro-Wilk* para testar a normalidade da amostra, pois apresenta melhor desempenho para discriminar a distribuição de amostra normal (RAZALI; WAH, 2011). Para realizar o teste de *Shapiro-Wilk*, foram definidas duas hipóteses (CAMPOS, 2000a; MORAES, 2016):

H₀: A amostra provém de uma distribuição normal;

H₁: A amostra não provém de uma distribuição normal.

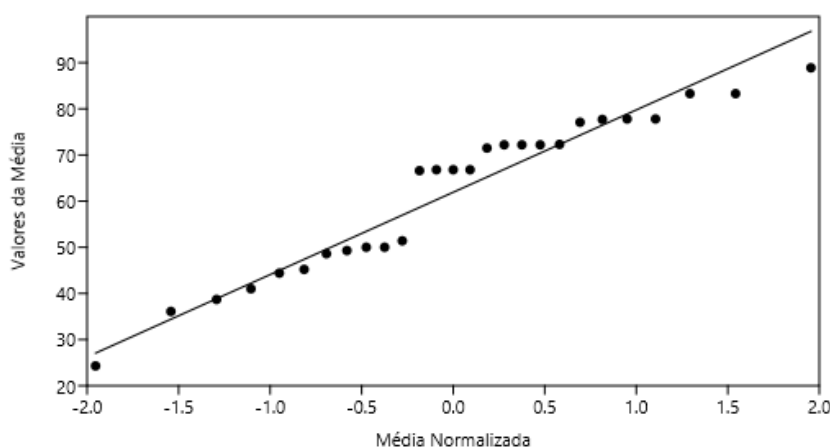
O valor resultante deste teste para o EC1 foi de $p=0,055$ e para o EC2 foi de $p=0,091$, considerando um $\alpha = 0,05$ (nível de significância). Logo, o valor de p (calculado) é maior que

⁷³ Disponível em: <https://folk.uio.no/ohammer/past/>. Acesso em: 15 mar. 2020.

⁷⁴ Disponível em: <https://sourceforge.net/projects/pspp4windows/>. Acesso em: 15 mar. 2020.

α , portanto não é possível rejeitar a H_0 . Dessa forma, se pode afirmar que para um nível de confiança de 95% (ou significância de 5%) a amostra provém de uma distribuição normal. A ilustração desta distribuição normalizada da amostra total ($N=27$) está apresentada no Gráfico 16, o qual revela que quanto mais perto os pontos estiverem do eixo central, maior a evidência da normalização dos dados.

Gráfico 16 - Gráfico da Distribuição Normal



Fonte: Da autora (2020).

Segundo Reis (2008, p. 277), “[...] sabe-se ou é razoável supor que a variável de interesse segue um modelo normal na população: isso significa que a distribuição amostral da média também será normal, permitindo realizar a inferência estatística paramétrica [...]” o que confirma o uso do teste realizado. O mesmo autor continua explicando:

[...] a distribuição da variável na população é desconhecida, mas a amostra retirada desta população é considerada “suficientemente grande”, o que, de acordo com o Teorema Central do Limite, permite concluir que a distribuição amostral da média é normal; e supõe-se também que a amostra é representativa da população e foi retirada de forma aleatória (REIS, 2008, p. 277-278).

Esse autor ainda fundamenta: “Há muita controvérsia a respeito do que seria uma amostra ‘suficientemente grande’, mas geralmente uma amostra com pelo menos 30 elementos costuma ser considerada grande o bastante para que a distribuição amostral da média possa ser aproximada por uma normal” (REIS, 2008, p. 278) e sugere o uso da distribuição t de *Student* para pequenas amostras.

Com os resultados obtidos e com a corroboração dos autores supracitados, se pode confirmar a utilização dos testes paramétricos. Segundo Campos (2000b), o teste de *Teste t (Student)* é o mais comumente utilizado na prática para duas amostras.

Considerando as seguintes hipóteses:

$$H_0: \mu_{EC1} = \mu_{EC2}$$

$$H_1: \mu_{EC1} \neq \mu_{EC2}$$

Esclarecendo que μ_{EC1} e μ_{EC2} correspondem às médias populacionais e não às médias amostrais, a Tabela 11 apresenta o resultado do *t Student* (amostras independentes) aplicado aos dados.

Tabela 11 - Teste *t* para duas amostras

	<i>EC1</i>	<i>EC2</i>
Média	59,426	62,656
Variância	192,769	330,345
Observações	6,000	21,000
Variância agrupada	302,830	
Diferença hipotética das médias	0,000	
Graus de Liberdade	25,000	
Estatístico <i>t</i>	-0,401	
P(T ≤t) unicaudal	0,346	
Valor crítico de <i>t</i> (unicaudal)	1,708	
P(T ≤t) bicaudal	0,692	
Valor crítico de <i>t</i> (bicaudal)	2,060	

Fonte: Da autora (2020).

Observando que o valor de $t_{\text{estatístico}}$ é igual a -0,401 e o valor crítico de *t* (bicaudal) é de 2,06, o primeiro valor se encontra no intervalo de -2,06 e +2,06, indicando uma distribuição normal. Usa-se o valor de *t* bicaudal ou bilateral, pois na hipótese H_1 foi considerado o sinal “≠” e portanto o valor de *p* deve corresponder às duas caudas da distribuição (BARBETTA, 2002).

Complementando, a regra $p(t \leq t_{\text{crítico}}) = 0,69$ e o nível de significância de $\alpha=0,05$ indicam que H_0 não é rejeitada. Em outras palavras, não há diferença significativa entre as médias. Logo, os dois Estudos de Caso foram realizados e o constructo andragógico não precisa ser alterado (bem desenvolvido em ambos os estudos).

Em suma, como não existe diferença significativa das médias entre as amostras populacionais, no passo seguinte os sujeitos foram reorganizados em dois grupos: brasileiros e

uruguayos, com o propósito de validar a conclusão da hipótese **d** da tese. Novamente, os dados foram analisados, conforme o fluxograma apresentado na Figura 58. A Tabela 12 mostra os dados descritivos.

Tabela 12 - Dados Descritivos de BR e UY

	Total	Média Final	Desvio Padrão (DP)	Coefficiente de Variação (CV)	Min	Max	Mediana	Moda
BR	12	58,08	17,65	30%	24,31	83,33	59,01	72,22
UY	15	65,02	16,63	26%	36,11	88,9	71,53	77,78

Fonte: Da autora (2020).

Esses dados (Tabela 12) apontam que a média da avaliação objetiva dos sujeitos uruguayos é maior com relação às dos brasileiros (em 11%), bem como o maior valor obtido (Max), a mediana e a moda desses sujeitos também é superior. Esses valores estão refletidos no Gráfico 14, no qual o número de sujeitos considerados ACod é maior para o grupo UY. No Apêndice K está detalhado o quantitativo das notas. Pode-se observar, ainda, que os valores que mais diferem entre os grupos são o mínimo (Min) e a mediana.

Para aplicar o teste de normalidade de *Shapiro-Wilk* no conjunto de dados, foram consideradas as seguintes hipóteses:

H₀: A amostra provém de uma distribuição normal;

H₁: A amostra não provém de uma distribuição normal.

Como resultado, obteve-se o valor de $p = 0,1498$. Então, sendo $p > \alpha$ e $\alpha = 0,05$, por certo não é possível rejeitar **H₀**, para um nível de significância de 5%; portanto, as amostras provém de uma população normal. O Gráfico 17 mostra esta distribuição, complementando com o histograma apresentado no Gráfico 18.

Gráfico 17 - Distribuição dos Dados de sujeitos BR e UY

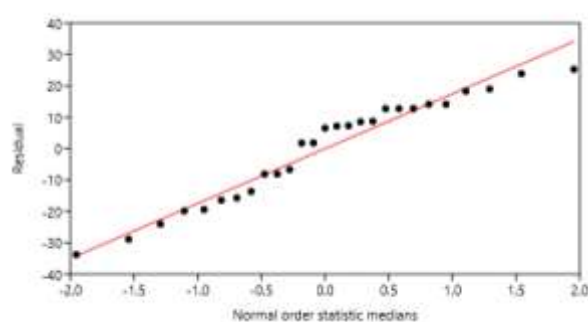
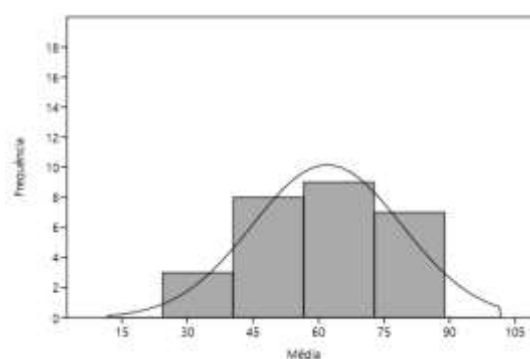


Gráfico 18- Histograma dos Dados de BR e UY



Fonte: Da autora (2020).

Para o formato do Gráfico 17, destaca-se que se os pontos se concentrarem em torno de uma reta, então tem-se indícios de que a distribuição é normal. Já para o histograma do Gráfico 18, o qual apresenta a forma da distribuição, ou seja, quanto mais próximo ao formato de sino, maiores as evidências de normalidade.

Calculando o teste *t* de *Student* para as duas amostras independentes, foram obtidos os resultados que estão apresentados na Tabela 13.

Tabela 13 - Teste *t* entre BR e UY

	BR	UY
Média	58,0828704	65,0222222
Variância	311,653964	276,443668
Observações	12	15
Variância agrupada	291,936199	
Diferença hipotética das médias	0	
Graus de Liberdade	25	
<u>Estatístico t</u>	-1,04864697	
<u>P(T<=t) unicaudal</u>	0,15218658	
<u>Valor crítico de t (unicaudal)</u>	1,70814076	

<u>P(T<=t) bicaudal</u>	0,30437317
Valor crítico de t (bicaudal)	2,05953855

Fonte: Da autora (2020).

Assumindo que as hipóteses são:

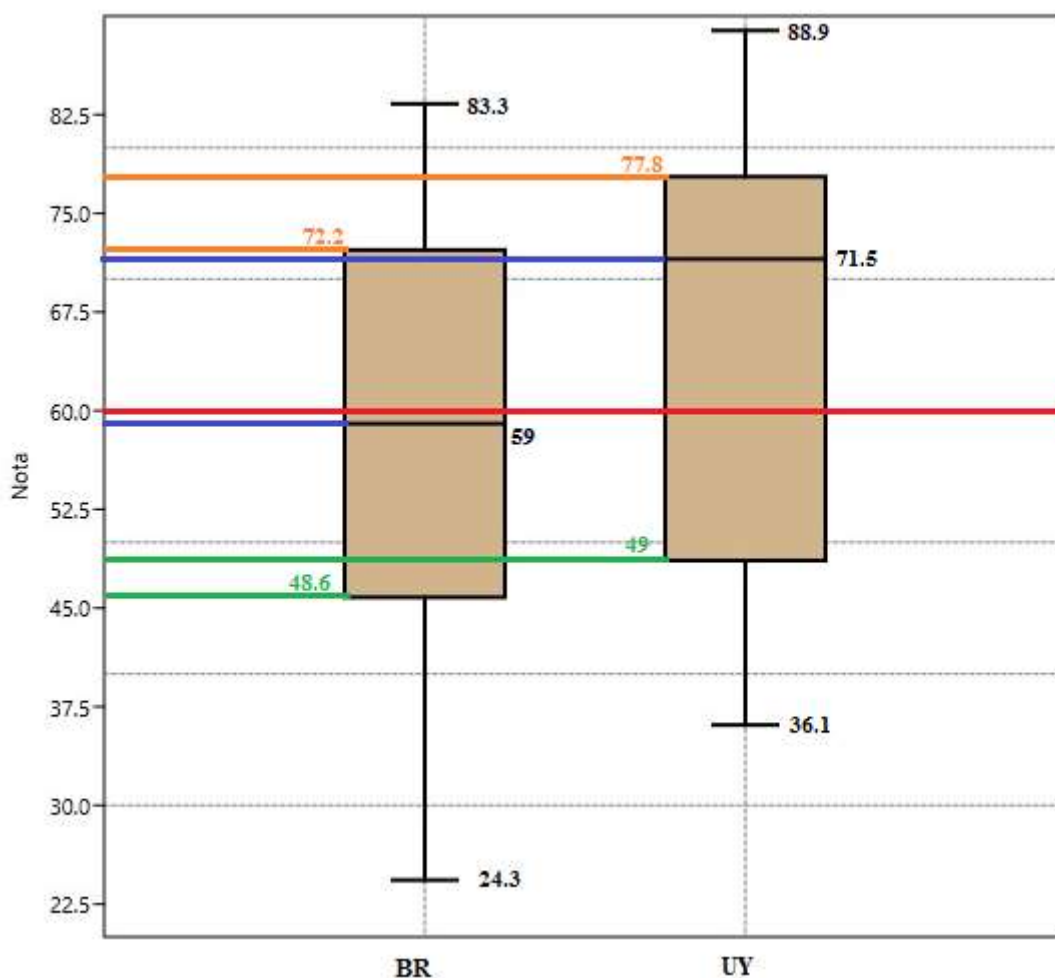
$$H_0: \mu_{BR} = \mu_{UY}$$

$$H_1: \mu_{BR} \neq \mu_{UY}$$

Observando que o valor de t é igual a -1.04 e o valor crítico de t (bicaudal) é de $2,05$, então o valor de t (estatístico) é menor que o crítico e $P(T \leq t)$ é igual a $0,30$. Logo, este valor é maior que o nível de significância de $\alpha=0,05$; portanto, a H_0 não é rejeitada. Consequentemente, não existe uma diferença significativa das médias populacionais entre os sujeitos brasileiros e uruguaios.

Cabe salientar que os dados sublinhados apresentados na Tabela 13 (na qual foram supostas variâncias iguais) apresentam o mesmo valor para o teste t quando se supõem variâncias diferentes. Portanto, pode-se inferir a mesma conclusão sobre ambos os casos, com relação à diferença do valor da variância das amostras de dados (BARBETTA, 2002).

A partir da observação do Gráfico 19, no formato de *Box Plot*, permite-se inferir as assimetrias dos dados (WICKHAM; STRYJEWSKI, 2011). Cabe salientar que este tipo de gráfico representa a distribuição da densidade dos dados (LEM *et al.*, 2014).

Gráfico 19- *Box Plot* dos sujeitos BR e UY

Fonte: Da autora (2020).

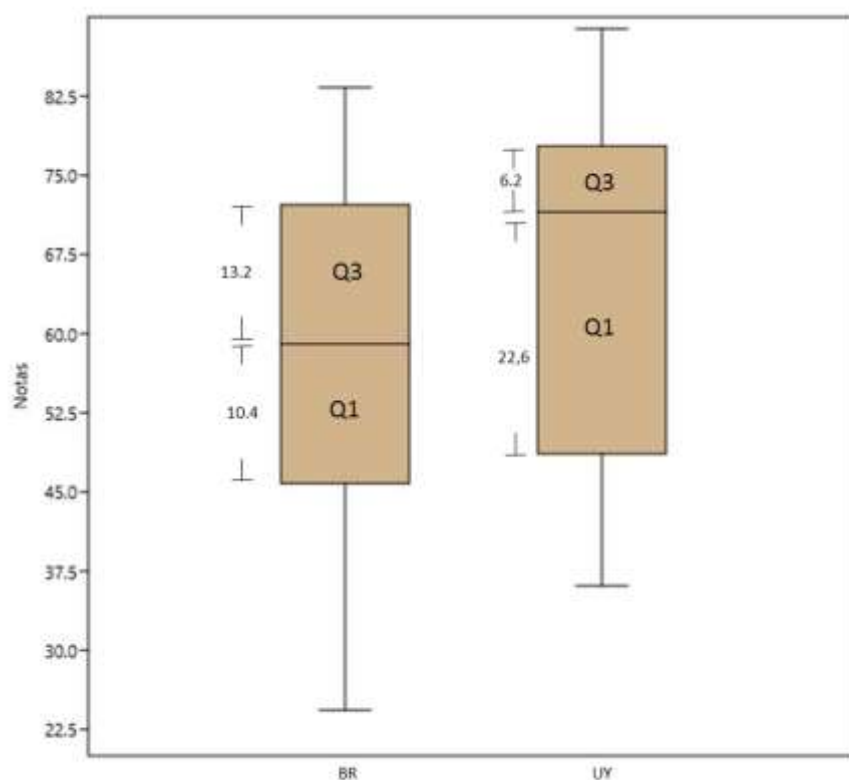
O conjunto de dados não apresentou nenhum valor atípico (*outliers*) (ANDERSON *et al.*, 2019). A mediana, ou quartil 2 (apresentado no gráfico pela cor azul), para a amostra de brasileiros, ficou bem próxima da média de referência do MRPC (representado pela cor vermelha). Em contrapartida, a mediana dos sujeitos uruguaios obteve o valor de 71,5, considerado superior ao valor de referência. Embora o interquartil, ou amplitude (quartil 3 – demonstrado na cor laranja – subtraído do quartil 1 – desenhado na cor verde) entre as amostras seja de somente 5,2, o grupo UY obteve a maior nota e o maior valor no quartil 3.

A assimetria da amostra BR se deu à direita (positiva), pois o valor do quartil superior subtraído da mediana foi igual a 13,2 que é maior do que 10,4, valor obtido da subtração da mediana pelo quartil inferior (Gráfico 20). O fato significa que “[...] o conjunto apresenta uma dispersão mais elevada nos valores maiores. Isso fez com que o quartil superior aumentasse de

valor (deslocando-o para a direita) e ficasse mais distante da mediana do que o inferior, significando assimetria para a direita (ou positiva)” (REIS, 2008, p. 117).

No entanto, o grupo dos uruguaios apresentou uma assimetria à esquerda, pois a diferença entre mediana e quartil inferior foi de 22,6 e de 6,2 para o quartil superior. O Gráfico 20 ilustra esses valores.

Gráfico 20 - Assimetrias



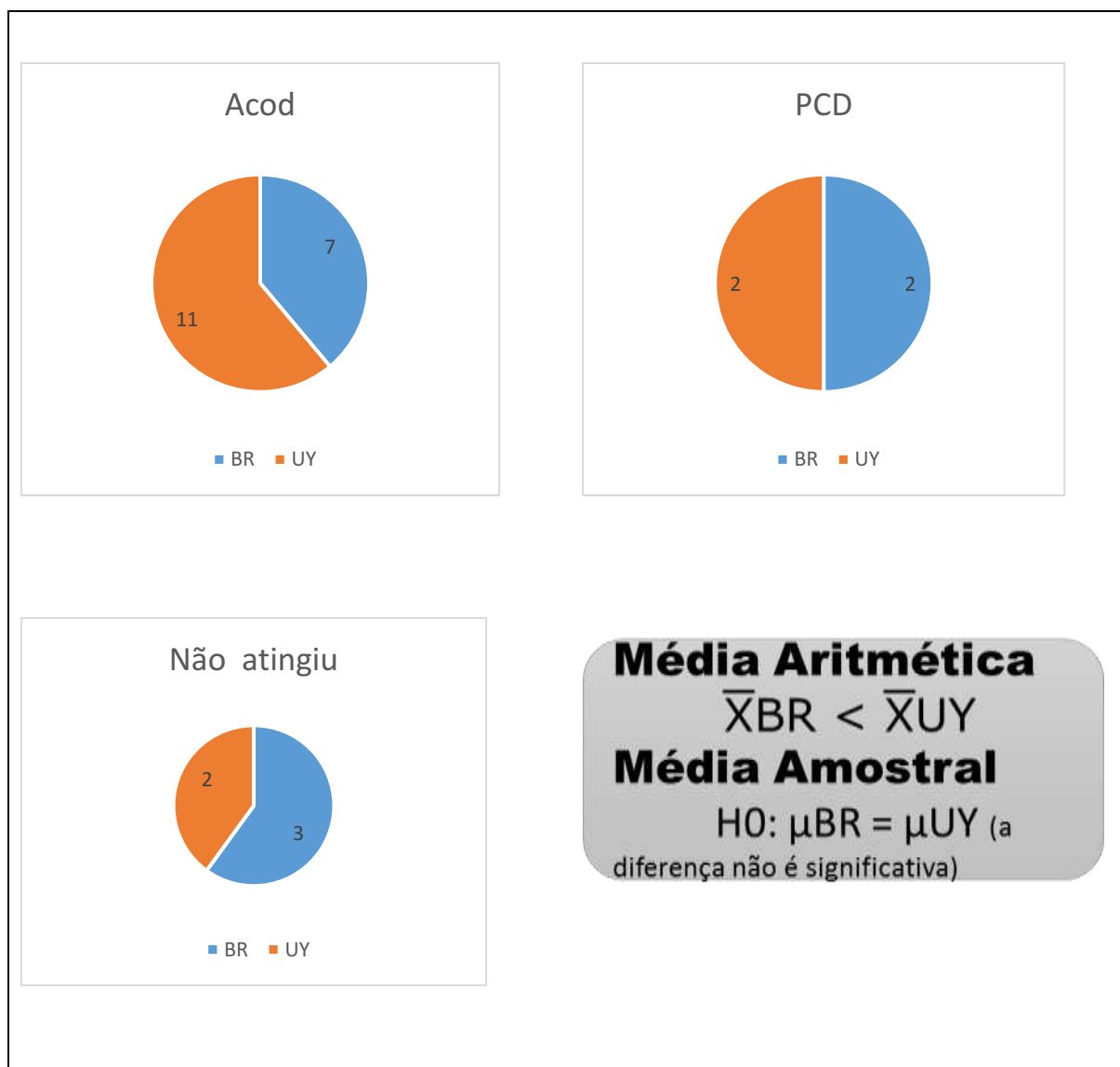
Fonte: Da autora (2020).

Concluindo, a amostra do grupo dos uruguaios foi levemente superior à do Brasil, porém estatisticamente não se diferem. Observa-se que, na amostra dos brasileiros, houve pouca variação, ou seja, os dados são mais homogêneos. Já nos uruguaios há maior concentração de sujeitos no Q1, porém poucos sujeitos com notas altas que elevaram a mediana (Q3). Os dados dos sujeitos UY foram mais discrepantes e heterogêneos, diferentemente da amostra brasileira.

O valor de referência para análise é de 6 (valor mínimo sugerido no MRPC). Pode-se concluir que no grupo BR houve uma menor variabilidade com relação aos sujeitos UY, ou seja, a mediana UY possui um valor superior com respeito à do grupo BR (mais homogêneo).

Com a intenção de resumir os resultados para responder à hipótese **d**, a Figura 59 é mostrada.

Figura 59 - Resumo da avaliação entre BR e UY



Fonte: Da autora (2020).

Retomando os dados, tem-se que a média amostral do grupo uruguaio foi maior que a amostra dos brasileiros ($65,02 > 58,08$), com uma dispersão no desvio padrão de 17,65 dos brasileiros. O quantitativo dos dados está demonstrado na Figura 59, ainda considerando que para a média amostral a H_0 não é rejeitada, ou seja, estatisticamente não há diferença entre as populações. Em outras palavras, a hipótese **d** da pesquisa⁷⁵ é rejeitada, ou seja, não há

⁷⁵ Hipótese D: há diferenças no resultado de formação do PC entre os sujeitos com formações diferentes de ensino, ou seja, entre brasileiros e uruguaios.

diferenças significativas entre o desenvolvimento do PC dos sujeitos da pesquisa (amostra de alunos dos cursos binacionais).

Com os dados referentes aos estudos de casos concluídos, ou seja, os dados da aplicação dos instrumentos desenvolvidos nesta pesquisa, fez-se um estudo sobre a relação entre variáveis (habilidades), como elas se comportam e a tendência a se alterarem de forma linear.

Na seção seguinte será analisada a correlação das variáveis referentes às habilidades avaliadas nos testes aplicados.

7.1 Coeficiente de Correlação e Determinação

A correlação, em estatística, avalia a associação entre duas variáveis do tipo qualitativas (REIS, 2008). Em outras palavras, é analisar **se** e **como** duas ou mais variáveis se relacionam entre si – correlação – seja na mesma ou em direções opostas (BARBETTA, 2002). O resultado da análise da correlação fornece um número que “[...] resume o grau de relacionamento entre duas variáveis [...]” (STEVENSON, 2001, p. 341) e continua que “em educação e psicologia, frequentemente se dá maior ênfase no grau ou força do relacionamento” (p. 341). Há outra técnica de estimativa relacionada à correlação que é a regressão, porém esta última não foi utilizada conforme esse mesmo autor, pois seu resultado retorna “uma equação que descreve o relacionamento em termos matemáticos” (p. 341), ou seja, uma equação de predição, muito utilizada em Administração e Economia, entre outras áreas.

No contexto desta tese, as correlações estão relacionadas às habilidades definidas em cada questão proposta nos testes realizados (pré-teste, teste 2 e pós-teste) e na avaliação das subcaracterísticas da ISO/IEC 25010 consideradas no Projeto Final. Retomando, no MRPC foram definidas cinco habilidades e salientando que uma questão de qualquer teste pode avaliar mais de uma habilidade.

A partir da correção de cada questão, fez-se o somatório de acertos de cada uma das habilidades para todos os sujeitos participantes dos Estudos de Casos. O resultado está resumido no Quadro 13, no qual H representa a habilidade avaliada e os sujeitos estão organizados por Estudo de Caso realizado.

Quadro 13 - Dados das habilidades dos sujeitos

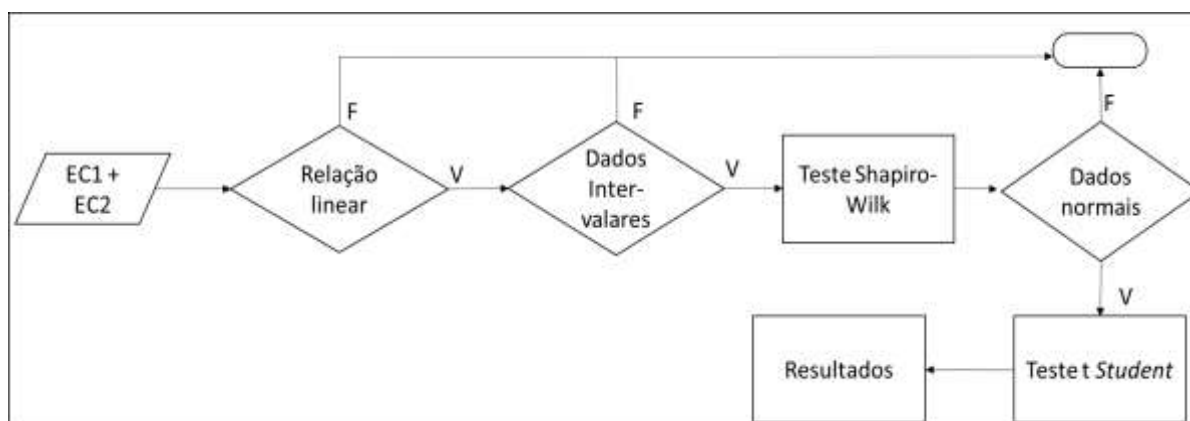
	ID aluno	H1	H2	H3	H4	H5
EC1	A02	5	6	6	4	4

	ID aluno	H1	H2	H3	H4	H5
	A04	4	4	4	3	3
	A06	5	5	6	4	4
	A08	4	4	5	3	3
	A09	6	4	4	1	1
	A10	2	2	1	0	0
EC2	A01	7	6	10	6	7
	A02	4	5	10	6	6
	A03	6	7	10	7	5
	A04	2	4	7	3	5
	A05	5	7	10	6	7
	A06	3	3	5	3	3
	A07	0	1	4	2	4
	A08	4	5	8	7	7
	A09	2	2	4	1	2
	A10	5	6	7	5	5
	A11	3	5	4	1	3
	A12	5	6	9	7	7
	A13	4	5	8	5	5
	A14	6	6	9	7	7
	A15	5	6	8	6	5
	A16	5	6	10	7	7
	A17	2	4	7	3	4
	A18	4	4	7	4	5
	A19	2	3	6	4	3
	A20	6	7	9	6	6
	A21	3	3	5	4	4

Fonte: Da autora (2020).

O coeficiente de correlação entre variáveis mais conhecido e utilizado é o “r de Pearson” (STEVENSON, 2001), por isso escolhido para analisar os dados da pesquisa. Segundo Vasconcelos e Feitosa (2017), há algumas exigências para o uso desse coeficiente, que são: a relação deve ser linear entre as variáveis, os dados devem ser intervalares, as características normalmente distribuídas e a amostragem aleatória (aplicação do teste de significância). Os passos executados podem ser acompanhados na Figura 60.

Figura 60 - Fluxograma das correlações



Fonte: Da autora (2020).

Para testar a linearidade dos dados e para cada correlação foi construído um diagrama de dispersão. Esses diagramas permitem a visualização da força, da direção e a natureza da correlação (REIS, 2008). Alguns desses gráficos são mostrados no decorrer do texto e os demais no Apêndice L. É importante pontuar que, em todos os gráficos construídos, a relação entre as variáveis ocorreu de forma linear. O segundo passo, explicado por Anderson *et al.* (2019, p. 21): “os dados intervalares são numéricos e expressos em termos de uma unidade de medida fixa”; portanto, contemplado esse requisito. O passo seguinte consistiu em verificar se a amostra é normalmente distribuída e assim poder aplicar o teste de significância. Salientam os mesmos autores que este último teste se faz necessário para amostras pequenas, ou seja, no qual N (número de participantes) seja menor que 30, no caso desta pesquisa, N=27.

Para o passo 3, o teste de *Shapiro-Wilk* foi utilizado e os resultados estão mostrados no Quadro 14.

Quadro 14 – Resultado do Teste de Normalidade

Habilidade	Valor (p)
H1	0,133
H2	0,109
H3	0,078
H4	0,052
H5	0,068

Fonte: Da autora (2020).

Para tanto, as seguintes hipóteses foram formuladas:

H₀: os dados estão normalizados

H₁: os dados não estão normalizados

Logo, tem-se que todos os $p(\text{valor})$ – vide Quadro 14 – são maiores que $\alpha=0,05$, ou seja, $p > 0,05$. Portanto H₀ não é rejeitado e pode-se afirmar que a amostra está normalmente distribuída.

Ainda se torna necessário executar o último passo, ou seja, testar a significância da medida de correlação, na qual se estabelecem as seguintes hipóteses:

H₀: não existe correlação entre as variáveis

H₁: existe correlação entre as variáveis.

Usando o teste t com distribuição bilateral (a correlação pode ser positiva ou negativa), e tendo o $t_{\text{critico}} = 2.006$, os resultados estão demonstrados no Quadro 15.

Quadro 15 - Teste t das Habilidades

t estatístico	H2	H3	H4	H5
H1	-1,42	-4,83	-0,43	-0,99
H2		-3,75	0,79	0,31
H3			4,05	3,79
H4				-0,47

Fonte: Da autora (2020).

Conforme a regra $P (T \leq t)$, se retornar verdadeira, portanto rejeita-se **H₀**, ou seja, não há diferença significativa. Ilustrado no Quadro 15, os valores que aparecem na cor vermelha significam que existe correlação entre as variáveis. Em contraste tem-se a Habilidade 3, que não rejeita **H₀**, logo há chance de que esses valores não sejam significativos, porém nada se pode afirmar com certeza.

Com todas as etapas realizadas, na sequência o Quadro 16 aponta os resultados do coeficiente r de Pearson de todas as variáveis (habilidades) do MRPC. Explicando, fez-se a correlação da H1 com as demais habilidades, da H2 com as demais, e assim sucessivamente.

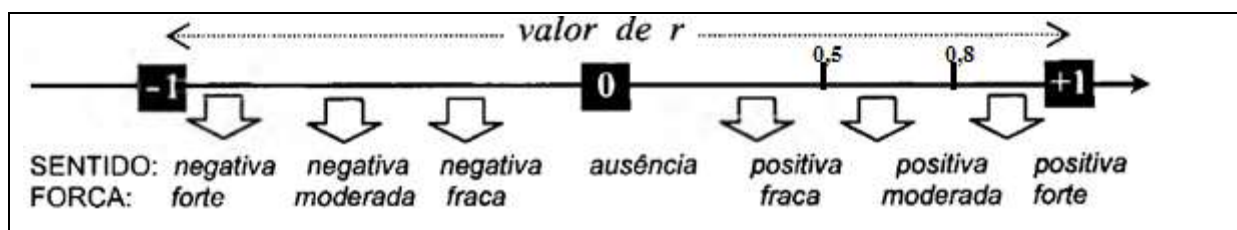
Quadro 16 - Coeficiente de Pearson

r de Pearson	H2	H3	H4	H5
H1	0,840	0,603	0,622	0,482
H2		0,790	0,765	0,694
H3			0,918	0,910
H4				0,905

Fonte: Da autora (2020).

A interpretação do valor de r de Pearson é ilustrada na Figura 61.

Figura 61 - Sentido e força da correlação em função do valor de r



Fonte: Adaptado pela autora, de Barbetta (2002, p. 278).

Para complementar a análise das correlações, aplicou-se o coeficiente de Determinação, representado por R^2 . Isto é, uma medida da proporção da variabilidade em uma variável que é explicada pela variabilidade da outra, obtido pela potência do coeficiente de r de Pearson (BARBETTA, 2002; ANDERSON *et al.*, 2019). Os valores desses coeficientes foram calculados e descritos no Quadro 17.

Quadro 17 - Coeficiente de Determinação

R^2	H2	H3	H4	H5
H1	84%	36%	39%	23%
H2		62%	59%	48%
H3			84%	83%
H4				82%

Fonte: Da autora (2020).

Fundamentando-se nos cálculos dos coeficientes apresentados até então, passa-se a analisá-los.

Com relação à H1 (Compreensão), o maior valor encontrado de correlação foi com a H2 (Abstração). Ou seja, o $r=0,84$ (coeficiente de Pearson apresentado no Quadro 16) indica uma forte e positiva correlação (conforme indicado na Figura 61). Outrossim, 84% (coeficiente de Determinação - Quadro 17) dos dados são correlacionados, ou seja, apenas 16% de H1 não pode ser explicada pela variabilidade de H2. Complementando, tem-se no Gráfico 21 o diagrama de dispersão entre H1 e H2 no qual aponta um relacionamento positivo entre as duas variáveis. Interpretando estas correlações no contexto de PC, significa considerar que quanto melhor a compreensão (H1) do problema, maior a chance de acertar a abstração (H2), ou seja, que essa última está relacionada com a coleta dos requisitos importantes para a resolução do problema, pois ambas, sob o ponto de vista do MRPC, estão fortemente relacionadas.

Ao se observarem os valores de r de Pearson entre H1 e H3 (Resolução de Problemas) e entre H1 e H4 (Resolução Algorítmica), a correlação ocorreu no sentido positivo, porém de força moderada. Já com respeito à R^2 , a variação de H1 somente está relacionada com 36% da variação de H3 e de 39% para H4. Entende-se que a compreensão influenciou a correta resolução de um problema, bem como sua escrita em formato de algorítmico e/ou programa, porém não de forma determinante. Cabe elucidar que todos os diagramas de dispersão construídos a partir dos valores dessas variáveis apresentam relacionamentos positivos.

Em contrapartida, observou-se uma fraca correlação entre H1 e H5 (Validação), conforme apresentam os valores de $r=0,482$ e $R^2=23\%$. Estes valores são os menores obtidos em todas as correlações; sendo assim, pode-se entender que a compreensão é a habilidade que menos influencia na validação dos dados e vice-versa. A fim de validar esse diagnóstico, apresenta-se o diagrama de dispersão no Gráfico 21.

Gráfico 21 - Diagrama de Dispersão entre H1 e H2

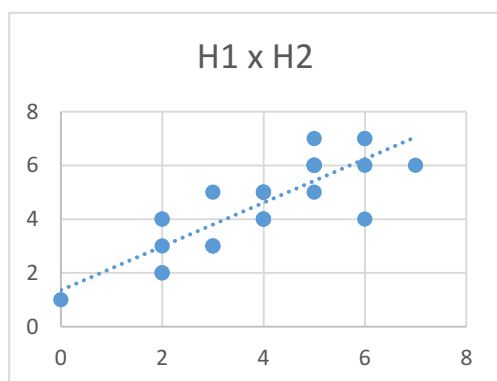
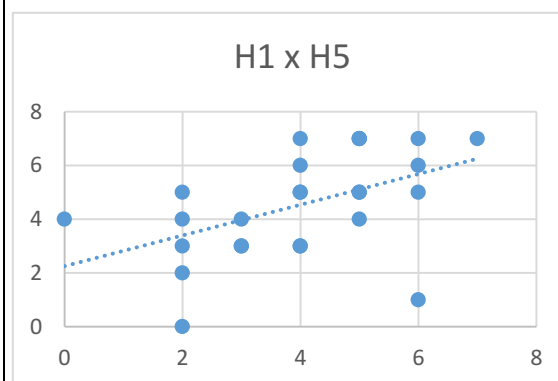


Gráfico 22 - Diagrama de Dispersão entre H1 e H5



Fonte: Da autora (2020).

Para as habilidades 1 e 5, o relacionamento entre ambas é considerado fraco, pois há maior dispersão dos mesmos (Gráfico 22).

As correlações da H2 com H3 e H4 apresentam valores próximos, respectivamente o coeficiente $r=0,79$ e $r=0,765$ apontam para uma correlação positiva e com força moderada. Para o coeficiente de Determinação, a variação entre H2 e H3 encontra-se em 62% de correlação, já entre H2 e H4 o valor obtido foi de 54% entre as mesmas. A mesma habilidade (Abstração) apresenta correlação moderada com a H5 (Avaliação), sendo o $r=0,694$ e o $R^2 = 48\%$ de variabilidade. Portanto, pode-se inferir que quanto mais correta a abstração melhor será a resolução do problema. Os diagramas de dispersão estão no Apêndice L.

Revisitando o Quadro 16, percebe-se que os maiores coeficientes r de Pearson ocorreram nas correlações de H3 com H4, pois $r=0,918$, e com uma pequena diferença entre H3 e H5, com o valor de $r=0,910$. Já para o coeficiente de determinação pode-se observar os valores de 84% e 83%, respectivamente. Analisando sob o ponto de vista do PC e do MRPC, pode-se depreender que a resolução correta do problema (H3) influencia, em 84%, na resolução algorítmica (H4) e na habilidade de avaliar (83%). Estendendo esta análise para o coeficiente de r de Pearson entre H4 e H5, ele apresentou um valor bastante próximo, ou seja, $r=0,905$ com $R^2 = 82\%$. Portanto, a correlação dessas três habilidades é positiva e forte. Os Gráficos 23 e 24 exibem o diagrama de dispersão entre H3 e H4 e de H3 e H5, respectivamente, confirmando, assim, a linearidade dos dados.

Gráfico 23 - Diagrama de Dispersão entre H3 e H4

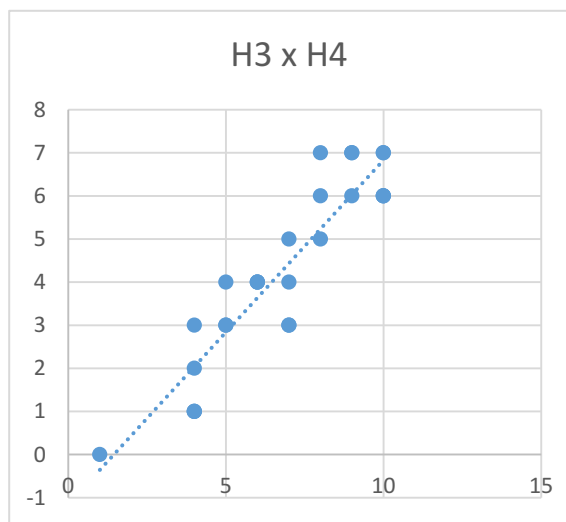
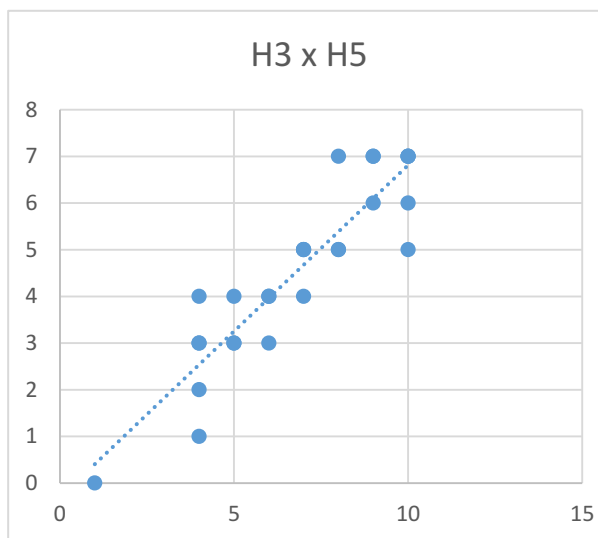


Gráfico 24 - Diagrama de Dispersão entre H3 e H5



Fonte: Da autora (2020).

Resumindo, a partir dos testes estatísticos é possível afirmar que todas as correlações foram positivas, variando de muito fraca (caso de H1 com H5) até o relacionamento positivo moderado a forte, com valores acima de +0,9 (STEVENSON, 2001). Ou seja, todas as habilidades apresentam correlação entre si, umas mais acentuadas, como no caso da habilidade de resolução do problema (H3), a qual seu valor aumenta, e na mesma proporção que aumentam H4 (84%) e H5 (83%).

Interpretando esses valores sob o ponto de vista de Pensamento Computacional: quando o sujeito compreende um problema consegue fazer as abstrações necessárias (H1 x H2, para $r=0,84$), assim conseguindo propor uma solução o mais correta possível (H2 com H3, onde $r=0,79$), bem como transformar essa solução em um algoritmo e/ou em um software (H3 x H4 onde $r=0,918$), sendo na sequência capaz de validá-lo ou testá-lo (H4 x H5 tendo $r=0,9$).

Portanto, o MRPC está correto e pode ser aplicado em outros estudos de caso, tendo a possibilidade de adicionar mais testes para atender a demanda e a realidade dos sujeitos adultos.

8 CONCLUSÃO

A expressão ‘Pensamento Computacional’ (PC) foi cunhada por Wing (2006) e, ao longo do tempo, a própria autora fez atualizações das ideias, sendo que nesta tese a nomenclatura foi entendida como sendo o “processo de pensamento envolvido na formulação de um problema e em expressar sua(s) solução(ões) de tal forma que um computador – humano ou máquina – possa efetivamente ser executado” (WING, 2017, p. 8, tradução nossa).

Porém, esse tema e outros como resolução de problemas, programação, raciocínio lógico podem ser encontrados nos trabalhos de Papert (1980a), com a sua teoria sobre o construcionismo. É certo que o PC trata sobre desenvolver habilidades cognitivas, entre elas a de resolver problemas utilizando-se de métodos e técnicas computacionais, mas não se restringe somente a isso.

Aliado a essa vertente, com as tecnologias arraigadas na vida cotidiana, pessoal e profissional, Rushkoff (2012) alerta para a necessidade das pessoas de passarem de consumidores de tecnologia para produtores de tecnologia a partir da idade escolar. No âmbito deste trabalho, os sujeitos escolhidos foram aqueles considerados adultos, pois muitos estudos estão tratando de pesquisar sobre o PC nas crianças e adolescentes, sendo este um dos motivos da escolha.

Este trabalho apresentou opiniões de diferentes autores sobre o entendimento de PC e algumas propostas de como avaliá-lo. Um estudo sobre o estado da arte sobre esses temas foi percorrido, através do qual pode-se perceber a escassez de pesquisas sobre o assunto em adultos, em comparação à preocupação de desenvolver o PC desde o Jardim de Infância até o término do Ensino Médio (e seus correspondentes no Uruguai). No nosso país, as pesquisas sobre esse tema ainda são incipientes.

A partir desses conhecimentos, aliados às questões de ACT, baseado nos vários tipos de alfabetizações (ou múltiplas alfabetizações) que um sujeito pode desenvolver, nesta tese apresentou-se uma definição de alfabetização de código (CORDENONZI *et al.*, 2020) e um método para a sua avaliação.

Com o intuito de responder ao objetivo da pesquisa, que foi de “compreender e analisar o desenvolvimento do PC nos sujeitos adultos com formação em escolas brasileiras e uruguaias”, foram propostos:

- a) um Modelo de Referência de Pensamento Computacional (MRPC);
- b) um construto andragógico para guiar o processo de ensino e aprendizagem dos sujeitos sobre PC, e
- c) método de avaliação.

No MRPC, foram definidas e descritas habilidades básicas que o sujeito pode desenvolver para pensar computacionalmente, além de propor a habilidade de Compreensão – inédita com referência aos autores pesquisados. No *framework* proposto por Brennan e Resnick (2012), os autores sugerem conceitos e práticas a serem desenvolvidas, porém muitos conceitos estão definidos e na prática utilizam o *Scratch* com o nível K-12. No MRPC, em comparação com o anterior, o conjunto de conceitos é menor, foi incluído o conceito de variáveis, e a prática pode ser programada em qualquer linguagem de programação.

Portanto, acredita-se que esta tese contribuiu em definir um modelo de referência de pensamento computacional e um constructo andragógico, tanto de ensino quanto de aprendizagem, que, aliados, classificam o sujeito ACod e avaliam o seu processo de construção sobre PC. A partir dos documentos desenvolvidos, o objetivo deste trabalho foi alcançado para cada aluno, de forma individual. Desse modo, o objetivo específico “analisar as habilidades e competências dos sujeitos com formação no Brasil/com formação no Uruguai, em uma situação de ensino e aprendizagem de PC” foi minuciosamente respondido no Capítulo 4 e validado nos Capítulos 5, 6 e 7.

Muitos foram os passos para se obter o resultado. Por meio do questionário “Conhecendo Você” e das observações realizadas pôde-se verificar o conhecimento prévio dos alunos, ou seja, seus subsunçores com relação ao tema proposto nesta tese, a fim de atender ao objetivo específico: “verificar o processo de alfabetização em código dos sujeitos”.

O *App Inventor* se mostrou uma adequada escolha de ferramenta para desenvolver a Habilidade 4, bem como para responder às expectativas dos alunos e do mercado de trabalho. A facilidade de sua aprendizagem e a velocidade em desenvolver e testar os App encantaram os alunos, pois rapidamente conseguiram perceber os resultados dos seus programas. Logo,

desenvolver a habilidade de programação, atualmente se torna importante, segundo Ben Pring, da *Cognizant* (empresa especializada em tecnologia)⁷⁶, pois afirma que de cada 13 novas profissões que irão surgir 10 exigirão conhecimento de linguagens de programação. Com base no que já foi exposto, a ACod e o LCod serão habilidades e competências diferenciais para os profissionais do futuro e, logicamente, para aqueles que já estão no mercado de trabalho e precisam se atualizar.

Importante pontuar que durante o desenvolvimento desta tese as ideias de David Ausubel e Malcolm Knowles estiveram sempre contempladas e foram as grandes influenciadoras deste processo. Já na escolha da ferramenta, mostrou-se na prática o domínio dos conceitos em consonância com o pensamento de Knowles, Holton e Swanson (2011), por exemplo, as escolhas dos temas de programação dos App, os quais devem sempre estar conectados com a realidade ou contexto dos aprendizes.

A habilidade Compreensão foi influenciada por Ausubel (1968), quando afirma que se deve sempre partir do que o aluno já sabe, complementado por testes, os quais devem avaliar diferentes aspectos do aprendiz, sendo um deles a compreensão, bem como aplicar os testes em formatos diferentes do que foram trabalhados em aula, a fim de verificar se a aprendizagem ocorreu, ou o aluno apenas decorou as respostas.

A decisão de não definir, previamente, o projeto final, ou seja, negociar com os sujeitos qual o problema a ser resolvido, embasou-se em Knowles (1981), quando afirma que o adulto sabe o que quer, tem prontidão e motivação para aprender (Princípios 4 e 6, respectivamente), além de torná-los partícipes do processo.

A validação dos instrumentos definidos foi realizada nos dois Estudos de Casos, apresentados, em detalhes, nos Capítulos 5 e 6. Propôs-se o desenvolvimento de Apps, com o intuito de serem utilizados nos dispositivos móveis dos usuários e, a partir deste ponto, acompanhar as habilidades desenvolvidas individualmente. Foi possível, a partir dos resultados obtidos, sugerir a eficácia da abordagem desenvolvida nesta tese. A complementação e a integração dos instrumentos avaliativos objetivos e subjetivos permitiram conhecer cada sujeito envolvido na pesquisa e observar sua aprendizagem. O empoderamento de conhecimentos demonstrados pelos alunos ficou evidente nos Estudos de Casos. Embora nem todos desenvolveram aplicativos suficientemente corretos e completos (sujeitos ACod), pôde-se perceber aqueles alunos que desenvolveram somente algumas habilidades de PC, mesmo que

⁷⁶ Revista Devmedia. Disponível em: <https://www.devmedia.com.br/profissoes-do-futuro-13-profissoes-promissoras-e-10-exigem-conhecimento-em-linguagens-de-programacao/39903>. Acesso em: 6 ago. 2019.

de forma desplugada. Acredita-se que a escolha da ferramenta *App Inventor* foi adequada tanto para os objetivos aos quais o curso se propôs, quanto às expectativas dos partícipes.

A partir da análise dos dados de forma qualitativa foram respondidas as hipóteses **b** e **c**, já a investigação quantitativa respondeu-se as hipóteses **a** e **d**. As habilidades propostas se mostraram relacionadas, conforme pôde ser percebido na análise de suas correlações. Portanto, sugere-se o uso do MRPC em outros estudos de casos. Em novas aplicações do MRPC o pesquisador pode aventurar-se a ampliar o tempo de aplicação e incluir um maior número de questões para os testes, sempre com o cuidado de identificar corretamente a qual habilidade está se referindo. Acredita-se que o arcabouço proposto neste trabalho poderá ser utilizado e/ou adaptado como uma disciplina a ser ofertada em cursos superiores, de qualquer área, como um fomento ao desenvolvimento do PC e ao uso das tecnologias digitais.

Parece importante uma reflexão sobre a pergunta que remeterá ao ponto de partida: por que este tema é importante? Acredita-se que aqui a lista de argumentos poderá ser bem extensa, mas alguns pensamentos poderão corroborar. Boa parte das pesquisas estão focadas em desenvolver o PC desde a idade do Jardim de Infância até o Ensino Médio (ou nível K-12, ou *bachillerato*, dependendo da estruturação curricular do país). Os adultos não podem ser esquecidos, pois não tiveram essa oportunidade anteriormente, e talvez seja um dever dos cursos de graduação ofertarem disciplinas que trabalhem o PC. Esses sujeitos estarão no mercado de trabalho (que está exigindo cada vez mais) em no máximo 5 anos. Nitidamente é uma questão mercadológica, mas que também não exclui uma função educacional, pois resolver problemas está além da programação. Caso não seja esta a questão, o adulto precisa saber fazer, ser independente – aprender a aprender. O sexto princípio, proposto por Knowles (1973), cita a motivação para aprender, seja essa oriunda do seu bem-estar pessoal, seja profissional. Portanto, é possível auxiliar o adulto a desenvolver o PC, segundo os resultados obtidos neste estudo, elevando seu *status* para “produtor de tecnologia”. O PC, dentre um de seus aspectos, abarca a resolução de problemas, que fomenta criatividade do indivíduo, gerando um *continuum* para a aprendizagem significativa (NOVAK, 1983) e, conseqüentemente, pode resultar em uma ideia ou produto inovador.

A fim de elucidar sobre a importância do tema, torna-se relevante citar as palavras de Rushkoff (2012, texto digital, tradução nossa):

A meu ver, a alfabetização em código é um requisito para a participação em um mundo digital. Quando adquirimos a linguagem, não aprendemos apenas a ouvir, mas também como falar. Quando adquirimos o texto, não aprendemos apenas a ler, mas também a escrever. Agora que temos computadores, estamos aprendendo a usá-los, mas não como programá-los. Quando não somos alfabetizados em código, devemos aceitar os dispositivos e *softwares* que usamos com quaisquer limitações que seus

criadores tenham embutido neles. Quantas vezes você alterou o conteúdo de uma lição ou uma apresentação porque não conseguiu descobrir como fazer a tecnologia funcionar da maneira que você queria?

Esse autor propõe uma reflexão sobre o valor da alfabetização em código, independentemente da idade, mas sim para o exercício da plena cidadania, bem como da vida pessoal e social. É libertar o sujeito para definir as suas necessidades e implementá-las utilizando a tecnologia.

Lindeman (1926), já interessado com a educação de adultos, definiu que esta é “processo por meio do qual os aprendizes se tornam conscientes da experiência significativa” e também entendida e renomeada para “educação ao longo de toda a vida”, conforme o entendimento da UNESCO, explicada por Delors *et al.* (1998, p. 100, tradução e grifo nosso):

[...] é a chave que abre as portas do século XXI e, bem além de uma adaptação necessária às exigências do mundo do trabalho, é a condição para um domínio mais perfeito dos ritmos e dos tempos da pessoa humana. [...] A educação ao longo de toda a vida torna-se assim, para nós, o meio de chegar a um equilíbrio mais perfeito entre trabalho e aprendizagem bem como ao exercício de uma **cidadania ativa**.

A educação de adultos pode e talvez deva ser um *continuum* educativo, coextensivo à vida e ampliado às dimensões de uma sociedade que se modifica constantemente. No Relatório da ONU *The Age of Digital Interdependence* (ONU, 2019), está discorrido sobre como será o gerenciamento das oportunidades e os riscos de uma rápida mudança tecnológica que impactará profundamente no futuro das pessoas e no futuro do planeta. E na junção desses dois temas, aliados com a situação das pessoas, especificamente no ano de 2020, atípico, pandêmico, essas questões se tornam realidade em um curtíssimo espaço de tempo. É preciso urgentemente buscar soluções.

REFERÊNCIAS

ALCALÁ, Adolfo. **Andragogía ciencia y arte de la educación de personas adultas**. 1. ed. Caracas: Universidad Nacional Abierta, 2010.

AMARAL, Érico Marcelo H. do. **Processo de ensino e aprendizagem de algoritmos integrando ambientes imersivos e o paradigma de blocos de programação visual**. 2015. Tese (Doutorado em Informática na Educação) - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2015.

ANDERSON, David R. *et al.* **Estatística aplicada à Administração e Economia**. 8. ed. São Paulo: Cengage, 2019.

ARAÚJO, Ana Liz Souto Oliveira de. **Quantifying Computational Thinking Abilities**. 2019. Tese (Doutorado em Ciência da Computação) - Universidade Federal de Campina Grande, Campina Grande, 2019.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. ABNT. **ISO/IEC 2382-1: Tecnologia da Informação – Vocabulário**. Rio de Janeiro: ABNT, 1998. Disponível em: <<https://www.iso.org/obp/ui/#iso:std:iso-iec:2382:-1:ed-3:v1:en>>. Acesso em: 12 mai. 2019.

AUSUBEL, David P. **Educational psychology: a cognitive view**. Holt, Rinehart and Winston: New York, 1968.

AUSUBEL, David P. **Aquisição e retenção de conhecimentos: Uma perspectiva cognitiva**. Tradução de Lígia Teopisto. 1. ed. Rio de Janeiro: Plátano Edições Técnicas, 2003.

AUSUBEL, David P.; NOVAK, Joseph Donald; HANESIAN, Helen. **Psicología Educativa** Un punto de vista cognoscitivo. 2. ed. México: Trillas, 1983.

AVELLO MARTÍNEZ, Raidell *et al.* Evolución de la alfabetización digital: nuevos conceptos y nuevas alfabetizaciones. **Medisur**, Cuba, v. 11, n. 4, p. 450–457, 2013. Disponível em: <http://scielo.sld.cu/pdf/ms/v11n4/ms09411.pdf>. Acesso em: 5 ago. 2018.

AVILA, Christiano *et al.* Metodologias de Avaliação do Pensamento Computacional:

uma revisão sistemática. *In*: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (SBIE), XXVIII., 2017, Recife. **Anais [...]**. Recife: SBC, 2017. Disponível em: <https://www.br-ie.org/pub/index.php/sbie/issue/view/171>. Acesso em: 10 fev. 2018.

BARBETTA, Pedro Alberto. **Estatística aplicada às Ciências Sociais**. 5. ed. Florianópolis, SC: UFSC, 2002.

BARBOSA, Lara Martins. **Aspectos do pensamento computacional na construção de fractais com o software GeoGebra**. 2019. Dissertação (Mestrado Profissional em Educação e Novas Tecnologias) - Universidade Estadual Paulista Júlio de Mesquita Filho, Rio Claro, 2019.

BARCELOS, Thiago Schumacher. **Relações entre o Pensamento Computacional e a Matemática em atividades didáticas de construção de jogos digitais**. 2014. Tese (Doutorado em Ensino de Ciências e Matemática) - Universidade Cruzeiro do Sul, São Paulo, 2014.

BARTON, David; HAMILTON, Mary. **Local Literacies: Reading and writing in one community**. 2012. Disponível em: https://www.academia.edu/12860191/Local_Literacies_Reading_and_Writing_in_One_Community. Acesso em: 14 ago. 2019.

BELLAN, Zezina. **Andragogia em ação - Como ensinar adultos sem se tornar maçante**. 6. ed. São Paulo: Z3 Editora e Livrarias, 2018.

BELSHAW, Doug. **The Essential Elements of Digital Literacies**, 2013. Disponível em: <https://doughelshaw.com/blog/2016/06/27/ebook-now-free/>. Acesso em: 30 ago. 2018.

BELSHAW, Doug. **Going Beyond “Learning to Code”**: Why 2014 is the Year of Web Literacy. 2014a. Disponível em: <https://clalliance.org/blog/going-beyond-learning-to-code-why-2014-is-the-year-of-web-literacy/>. Acesso em: 10 ago. 2018.

BELSHAW, Doug. **Why the Web Literacy Map will remain at v1.1 until MozFest**. 2014b. Disponível em: <http://literaci.es/weblitmap-at-v11-until-mozfest>. Acesso em: 12 set. 2018.

BETTS, Ryan. Concept to Code. **UX Magazine**, Colorado, USA, 2011. Disponível em: <http://uxmag.com/articles/concept-to-code>. Acesso em: 10 ago. 2018.

BOCHECO, Otávio. **Parâmetros para abordagem de evento no enfoque CTS**. 2011. Dissertação (Mestrado em Educação Científica e Tecnológica) - Universidade Federal de Santa Catarina, Florianópolis, 2011.

BONILLA, Maria Helena. O Brasil e a alfabetização digital. **Jornal da Ciência**, Rio de Janeiro, p. 7, 2011. Disponível em: <http://www2.ufba.br/~bonilla/artigojc.htm>. Acesso em: 15 ago. 2018.

BOZOLAN, Sandra Muniz. **O pensamento computacional: ensino e aprendizagem através do software Processing**. 2016. Dissertação (Mestrado em Tecnologias da Inteligência e Design Digital) - PUC-SP, São Paulo, 2016.

BRACKMANN, Christian Puhmann. **Desenvolvimento do pensamento computacional através de atividades desplugadas na Educação Básica**. 2017. Tese (Doutorado em Informática na Educação) - Universidade Federal do Rio Grando do Sul, Porto Alegre, 2017.

BRASIL. Portaria N° 125, de 21 de março de 2014. Estabelece o conceito de cidades-gêmeas nacionais, os critérios adotados para essa definição e lista todas as cidades brasileiras por estado que se enquadram nesta condição. **Diário Oficial**, Brasília, DF, 21 mar. 2014. Seção 1, p. 45. Disponível em: http://www.lex.com.br/legis_25369237_PORTARIA_N_125_DE_21_DE_MARCO_D#:~:text=Estabelece%20o%20conceito%20de%20cidades,que%20se%20enquadram%20nesta%20condi%C3%A7%C3%A3o. Acesso em: 13 jul. 2019.

BRENNAN, Karen; RESNICK, Mitchel. New frameworks for studying and assessing the development of computational thinking. **American Educational Research Association - AERA**, Washington, EUA, p. 1–25, 2012. Disponível em: https://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf. Acesso em: 10 maio. 2018.

BROWN, Wayne. **Introduction to algorithmic thinking**. 2015. Disponível em: [https://raptor.martincarlisle.com/Introduction to Algorithmic Thinking.doc](https://raptor.martincarlisle.com/Introduction%20to%20Algorithmic%20Thinking.doc). Acesso em: 03 jan. 2019.

CAJAS, Fernando. Alfabetización científica y tecnológica: la transposición didáctica del conocimiento tecnológico. **Enseñanza de las Ciencias**, Barcelona, v. 19, n. 2, p. 243–253, 2001. Disponível em: <https://www.raco.cat/index.php/Ensenanza/article/view/21737>. Acesso em: 6 ago. 2018.

CAMPOS, Geraldo Maia. **Interpretação dos resultados**. 2000a. Disponível em: http://www.forp.usp.br/restauradora/gmc/gmc_livro/gmc_livro_cap15.html. Acesso em: 1 abr. 2020.

CAMPOS, Geraldo Maia. **A escolha do teste mais adequado**. 2000b. Disponível em: http://www.forp.usp.br/restauradora/gmc/gmc_livro/gmc_livro_cap14.html. Acesso em: 1 abr. 2020.

CARVALHO, Jair Antonio de *et al.* Andragogia: Considerações sobre a aprendizagem do adulto. **Ensino, Saúde e Ambiente**, Niterói, RJ, v. 3, n. 1, p. 78–90, 2010.

CARVALHO, Felipe José R. **Introdução à programação de computadores por meio de uma tarefa de modelagem matemática na educação matemática**. 2018. Dissertação (Mestrado em Ensino) - Universidade Estadual do Oeste do Paraná, Foz do Iguaçu, 2018.

CASTILLO, Flavio de Jesús. **Las ciencias agógicas y su importancia en la actualidad**. 2018a. Disponível em: <http://otrasvoceseneducacion.org/archivos/268427>. Acesso em: 30 maio. 2019.

CASTILLO, Flavio de Jesús. Andragogía, andragogos y sus aportaciones. **Voces de la educación**, Xalapa, México, v. 3, n. 6, p. 64–76, 2018b. Disponível em: <https://www.revista.vocesdelaeducacion.com.mx/index.php/voces/article/view/120>. Acesso em: 2 maio. 2019.

CAVALCANTE, Ahemenson; COSTA, Leonardo dos Santos; ARAUJO, Ana Liz. Um estudo de caso sobre competências do pensamento computacional desenvolvidas na programação em blocos no Code.Org. *In: WORKSHOPS DO CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (CBIE), V.*, 2016, Uberlândia, MG. **Anais [...]**. Uberlândia, MG: SBC, 2016. Disponível em: <https://www.br-ie.org/pub/index.php/wcbie/article/view/7037>. Acesso em: 3 mar. 2018.

CAVALCANTI, Roberto de Albuquerque. Andragogia: A aprendizagem nos adultos. **Revista de Clínica Cirúrgica da Paraíba**, João Pessoa, v. 6, n. 4, p. 3–9, 1999. Disponível em: <https://pt.slideshare.net/Vicentana/andragogia-a-aprendizagem-nos-adultos>. Acesso em: 3 maio. 2019.

CAVALCANTI, Roberto de Albuquerque; GAYO, Maria Alice Fernandes da Silva. Andragogia na educação universitária. **Conceitos**, João Pessoa, PB, p. 44–51, 2005.

CERF, Vinton G. The Science in Computer Science. Computer science revisited. **Communications of the ACM**, New York, p. 4, 2012. Disponível em: <http://dl.acm.org/citation.cfm?doid=2380656.2380658>. Acesso em: 12 ago. 2018.

CHEMIN, Beatris F. **Manual da Univates para trabalhos acadêmicos**. 4. ed. Lajeado: Univat, 2020. Disponível em: <https://www.univates.br/editora-univates/publicacao/315>. Acesso em: 22 jul. 2020.

CITOLIN, Cristina B. **Eu Falo, Tu Hablas, Vos Hablás, Nós Ensinamos e Aprendemos Juntos: Aulas de Línguas em Cursos Binacionais**. 2013. Tese (Doutorado em Educação) - Universidade do Vale do Rio dos Sinos, São Leopoldo, RS, 2013.

COELHO, Marcos Antônio; DUTRA, Lenise Ribeiro; MARIELI, Joane. Andragogia e heutagogia: práticas emergentes na educação. **Revista Transformar**, Itaperuna, RJ, v. 8, n. 8, p. 97–107, 2016. Disponível em: <http://www.fsj.edu.br/transformar/index.php/transformar/article/view/87>. Acesso em: 2 set. 2019.

CORDENONZI, Walkiria *et al.* **Mapeamento da Educação Binacional - Relatório Técnico I**. Santana do Livramento, RS, 2014. Disponível em: http://ecompartindo.santana.ifsul.edu.br/pluginfile.php/2328/mod_resource/content/3/relatorio_1_-_mapeamento_da_educacao_binacional.pdf. Acesso em: 3 mar. 2019.

CORDENONZI, Walkiria Helena; DEL PINO, José Claudio; OLIVEIRA, Eniz Conceição; STROHSCHOEN, Andreia Aparecida Guimarães. Alfabetização – uma evolução do conceito: alfabetização e letramento em código. **Texto Livre: Linguagem e Tecnologia**, Belo Horizonte, MG, v. 13, n. 1, p. 137, 2020. Disponível em: <http://www.periodicos.letras.ufmg.br/index.php/textolivre/article/view/16133>. Acesso em: 26 maio. 2020.

CORNELL UNIVERSITY. **Cornell University Digital Literacy resource: Digital literacy is...** 2015. Disponível em: <https://digitalliteracy.cornell.edu/>. Acesso em: 5 ago. 2018.

COUTO, Natália Ellery Ribeiro. **Impacto da implantação de uma disciplina de pensamento computacional em estudantes - Um estudo de caso**. 2019. Dissertação (Mestrado Acadêmico de Computação Aplicada) - Universidade do Vale do Itajaí,

Itajaí, 2019.

CROVADOR, Alvaro. **O uso do Arduíno em sala de aula no experimento do pêndulo simples de Galileu Galilei**. 2019. Dissertação (Mestrado Profissional em Educação e Novas Tecnologias) - Centro Universitário Internacional - UNINTER, Curitiba, Paraná, 2019.

CRTC. **Currículo de Referência em Tecnologia e Computação**. São Paulo, SP.2018. Disponível em: <http://curriculo.cieb.net.br/>. Acesso em: 19 mar. 2019.

CUNHA, Marcos Ribeiro. **Gestão estratégica de IES: Modelos e funções do planejamento estratégico em universidades públicas e privadas de Palmas - Tocantins**. 2011. Dissertação (Mestrado em Educação) - Universidade Lusófona de Humanidades e Tecnologias, Lisboa, 2011.

CUNHA, Rodrigo Bastos. O que significa alfabetização ou letramento para os pesquisadores da educação científica e qual o impacto desses conceitos no ensino de Ciências. **Ciência & Educação**, Bauru, v. 24, n. 1, p. 27–41, 2018. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1516-73132018000100027&lng=pt&tlng=pt. Acesso em: 1 set. 2018.

DAGIENÉ, Valentina; FUTSCHEK, Gerald. Bebras International Contest on Informatics and Computer Literacy: Criteria for Good Tasks. *In: INFORMATICS EDUCATION - SUPPORTING COMPUTATIONAL THINKING - INTERNACIONAL CONFERENCE ON INFORMATICS, III .*, 2008, Torun, Polônia. **Anais [...]**. Torun, Polônia: Springer, 2008.

DAGIENÉ, Valentina; SENTANCE, Sue; STUPURIENÉ, Gabrielė. Developing a two-dimensional categorization system for educational tasks in informatics. **Informatica**, Vilnius, Lituânia, v. 28, n. 1, p. 23–44, 2017. Disponível em: <https://www.mii.lt/Informatica/htm/INFO1127.htm>. Acesso em: 28 mar. 2019.

DEL PINO, José Claudio; FRISON, Marli Dallagnol. Química: um conhecimento científico para a formação o cidadão. **Revista de Educação, Ciências e Matemáticas**, Rio de Janeiro, v. 1, n. 1, p. 36–50, 2011. Disponível em: <http://publicacoes.unigranrio.edu.br/index.php/recm/article/view/1585>. Acesso em: 10 set. 2018.

DELORS, Jacques *et al.* **Educação: um tesouro a descobrir**. Relatório para a UNESCO da Comissão Internacional sobre Educação para o Séc. XXI. 1. ed. São Paulo: Cortez, 1998.

DENNING, Peter. Ubiquity symposium: The science in computer science. Ubiquity, [S.l.], v. 2012, n. December, p. 1–5, 2012. Disponível em: <https://doi.org/10.1145/2406356.2406357>. Acesso em: 2 jul. 2018.

DIAS, Henrique. **Aprenda a programar - Uma breve introdução**. [S.l.:s.n.]: E-book, 2015. Disponível em: <https://pplware.sapo.pt/tutoriais/aprenda-a-programar-o-e-book-gratuito-e-essencial/>. Acesso em: 10 maio. 2018.

DUFVA, Tomi. **Code Literacy**. Understanding the programmed world. Finlândia: Aalto University, 2013. Disponível em:

<https://aaltodoc.aalto.fi/handle/123456789/11380>. Acesso em: 13 set. 2018.

FERNANDES, Hugo B. **Pensamento Computacional**: uma proposta de curso de extensão para professores que lecionam Matemática nos anos iniciais do Ensino Fundamental. 2018. Dissertação (Mestrado em Ensino de Ciências e Matemática) - Universidade Cruzeiro do Sul, São Paulo, 2018.

FONSECA, João José Saraiva da. **Metodologia da Pesquisa Científica**. Fortaleza: UECE - Universidade Estadual do Ceará, 2002.

FRANÇA, Rozelma Soares de; TEDESCO, Patrícia C. A. R. Explorando o pensamento computacional no Ensino Médio: do design à avaliação de jogos digitais. *In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), XXIII, 2015, Recife, PE. Anais [...]*. Recife, PE. Disponível em: <http://www.lbd.dcc.ufmg.br/bdbcomp/servlet/Trabalho?id=23409>. Acesso em: 22 nov. 2018.

FREITAS, Maria Teresa. Letramento digital e formação de professores. **Educação em Revista**, Belo Horizonte, v. 26, n. 3, p. 335–352, 2011. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0102-46982010000300017&lng=pt&tlng=pt. Acesso em: 25 ago. 2018.

FROTA, Vitor Bremgartner da. **Arcabouço conceitual de adaptação de recursos educacionais**. 2017. Tese (Doutorado em Informática) - Universidade Federal do Amazonas, Amazonas, 2017.

GARCIA, Carlos Henrique. Tabelas para classificação do coeficiente de variação. **IPEF**, Piracicaba, n. 171, p. 10, 1989.

GARCIA, Lenise Aparecida Martins. Competências e habilidades: você sabe lidar com isso. **Educação e Ciência on-line**, Brasília, 2005. Disponível em: http://www.educacao.es.gov.br/download/roteiro1_competenciasehabilidades.pdf. Acesso em: 15 maio. 2020.

GATTI, Bernardete A. Estudos quantitativos em Educação. **Educação e Pesquisa**, São Paulo, v. 30, n. 1, p. 11–30, 2004.

GATTI, Bernardete A. A construção metodológica da pesquisa em educação : desafios. **RBPAE**, Porto Alegre, v. 28, n. 1, p. 13–34, 2012.

GIL, Antonio C. **Como elaborar projetos de Pesquisa**. 4. ed. São Paulo: Atlas, 2002.

GIL, Antonio C. **Métodos e Técnicas da Pesquisa Social**. 6. ed. São Paulo: Atlas, 2008.

GOMES, Tancicleide C. S.; MELO, Jeane C. B. de. O pensamento computacional no Ensino Médio: Uma abordagem Blended Learning. *In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, XXI, 2013, Maceió. Anais [...]*. Maceió: SBC, 2013. Disponível em: <https://docs.google.com/file/d/0ByMLHgMMHCODM2h0UFRPa0tiZEE/edit>. Acesso em: 10 out. 2018.

GONÇALVES, Filipe Augusto. **Um instrumento para o diagnóstico do pensamento computacional**. 2015. Dissertação (Mestrado em Computação Aplicada) - Universidade do Vale do Itajaí, Itajaí, 2015.

GOOGLE. **O blog do Google Brasil: Melhora nas habilidades digitais pode transformar a vida de milhares de brasileiros**. 2019. Disponível em: <https://brasil.googleblog.com/2019/03/melhora-nas-habilidades-digitais-pode.html>. Acesso em: 9 ago. 2019.

GOOGLE FOR EDUCATION. **Google for Education: Computational Thinking**. 2015. Disponível em: <https://edu.google.com/resources/programs/exploring-computational-thinking/>. Acesso em: 20 mar. 2019.

GOUWS, Lindsey; BRADSHAW, Karen; WENTWORTH, Peter. First year student performance in a test for computational thinking. *In: PROCEEDINGS OF THE SOUTH AFRICAN INSTITUTE FOR COMPUTER SCIENTISTS AND INFORMATION TECHNOLOGISTS CONFERENCE ON - SAICSIT*, 13, 2013, New York, USA. **Anais [...]**. New York, USA: ACM Press, 2013. Disponível em: <http://dl.acm.org/citation.cfm?doid=2513456.2513484>. Acesso em: 10 abr. 2019.

GREFF, Guaraci *et al.* Aprendizagens em movimento: Um experimento de estímulo ao pensamento computacional de docentes com M-Learning e U-Learning. **Revista Thema**, Pelotas, v. 15, n. 1, p. 312–322, 2018. Disponível em: <http://periodicos.ifsul.edu.br/index.php/thema/article/view/536>. Acesso em: 12 abr. 2019.

GROVER, Shuchi; PEA, Roy. Computational Thinking in K–12. **Educational Researcher**, Thousand Oaks, EUA, v. 42, n. 1, p. 38–43, 2013. Disponível em: <http://journals.sagepub.com/doi/10.3102/0013189X12463051>. Acesso em: 19 mar. 2019.

HASE, Stewart; KENYON, Chris. From Andragogy to Heutagogy. **ultiBASE**, Melbourne, Austrália, v. 28, 2000. Disponível em: https://epubs.scu.edu.au/gcm_pubs/99. Acesso em: 2 set. 2019.

ISTE. **ISTE Standards for StudentsIste. Org/Standards**. [S.l.]: International Society for Technology in Education, 2016. Disponível em: <https://www.iste.org/standards/for-students>. Acesso em: 25 nov. 2018.

ISTE; CSTA. **Operational Definition of Computational Thinking for K–12 Education**. 2011. Disponível em: <https://www.iste.org/explore/Solutions/Computational-thinking-for-all>. Acesso em: 20 mar. 2019.

KALELIOGLU, Filiz; GÜLBAHAR, Yasemin; KUKUL, Volkan. A framework for computational thinking based on a systematic research review. **Baltic Journal of Modern Computing**, Riga, Letônia, v. 4, n. 3, p. 583, 2016. Disponível em: <https://draweb.njcu.edu:2052/central/docview/1810312661/abstract/42E985869B844290PQ/1?accountid=12793>. Acesso em: 4 mar. 2019.

KAMPFF, Adriana J. C. *et al.* Pensamento Computacional no Ensino Superior: Relato de uma oficina com professores da Universidade do Vale do Rio dos Sinos. *In:*

WORKSHOPS DO V CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO 2016, Uberlândia. **Anais [...]**. Uberlândia: SBC, 2016. Disponível em: <http://br-ie.org/pub/index.php/wcbie/article/view/7057>. Acesso em: 10 abr. 2019.

KATAI, Zoltan; TOTH, Laszlo. Technologically and artistically enhanced multi-sensory computer-programming education. **Teaching and Teacher Education**, [S.l.], v. 26, n. 2, p. 244–251, 2010. Disponível em: <http://dx.doi.org/10.1016/j.tate.2009.04.012>. Acesso em: 13 set. 2018.

KLEIMAN, Angela C. B. R. Modelos de letramento e as práticas de alfabetização na escola. In: KLEIMAN, Angela. B. (org.). **Os significados do letramento**. Novas perspectivas sobre a prática social da escrita. 1. ed. São Paulo. p. 15–61.

KNOWLES, Malcolm S. **The adult learner: A neglected species**. Houston: Gulf, 1973.

KNOWLES, Malcolm S. **The modern practice of adult education: From pedagogy to andragogy**. 2. ed. Nova York: Cambridge Book, 1981.

KNOWLES, Malcolm S.; HOLTON, Elwood F.; SWAMSON, Richard A. **Aprendizagem de Resultados**. 2. ed. São Paulo: Elsevier, 2011.

KNOWLES, Malcolm S.; HOLTON, Elwood F.; SWANSON, Richard A. **The adult learner - The definitive classic in adult education and human resource development**. 6. ed. San Diego: Elsevier, 2005.

KOH, Kyu Han *et al.* Towards the automatic recognition of computational thinking for adaptive visual language learning. In: SYMPOSIUM ON VISUAL LANGUAGES AND HUMAN-CENTRIC COMPUTING (VL/HCC), 2010, Madrid. **Anais [...]**. Madrid: IEEE, 2010. Disponível em: <http://ieeexplore.ieee.org/document/5635189/>. Acesso em: 26 mar. 2019.

KORDAKI, Maria. A drawing and multi-representational computer environment for beginners' learning of programming using C: Design and pilot formative evaluation. **Computers and Education**, [S.l.], v. 54, n. 1, p. 69–87, 2010.

KORKMAZ, Özgen; ÇAKIR, Recep; ÖZDEN, M. Yaşar. A validity and reliability study of the computational thinking scales (CTS). **Computers in human behavior**, [S.l.], v. 72, p. 558–569, 2017.

KOTSOPOULOS, Donna *et al.* A pedagogical framework for computational thinking. **Digital Experiences in Mathematics Education**, [S.l.], v. 3, n. 2, p. 154–171, 2017. Disponível em: <https://link.springer.com/article/10.1007/s40751-017-0031-2>. Acesso em: 22 maio. 2020.

KRATHWOHL, David R. A revision of Bloom's Taxonomy: an overview. **Theory into Practice**, [S.l.], v. 41, n. 4, p. 212–218, 2002. Disponível em: <http://www.jstor.org/stable/1477405>. Acesso em: 30 nov. 2018.

LAISA, Jéssica; HENRIQUE, Eduardo; OLIVEIRA, Wendell. Idealizando jogos digitais de pensamento computacional a partir do Bebras Challenge: Um estudo exploratório. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 26., 2018, Natal, RN. **Anais [...]**. Natal, RN: SBC, 2018.

- LAKATOS, Eva M.; MARCONI, Mariana de A. **Fundamentos de Metodologia Científica**. 5. ed. São Paulo: Atlas, 2003.
- LEM, Stephanie *et al.* Experts' Misinterpretation of Box Plots - a Dual Processing Approach. **Psychologica Belgica**, [S.l.], v. 54, n. 4, p. 395–405, 2014.
- LI, Ying. Research into the computational thinking for the teaching of computer science. *In*: FRONTIERS IN EDUCATION CONFERENCE, FIE 2014, Madrid. **Anais [...]**. Madrid: IEEE, 2014. Disponível em: <https://ieeexplore.ieee.org/document/7044465>. Acesso em: 30 out. 2018.
- LINDEMAN, Eduard C. **The Meaning of Adult Education**. 1. ed. Nova York: New Republic. E-book, 1926. Disponível em: https://openlibrary.org/books/OL14361073M/The_meaning_of_adult_education. Acesso em: 22 maio. 2020.
- LUMMERTZ, Ramon dos Santos. **As Potencialidades do Uso do Software Scratch para a Construção da Literacia Digital**. 2016. Dissertação (Mestrado em Ensino de Ciências e Matemática) - Universidade Luterana do Brasil, Canoas, 2016.
- LYE, Sze Yee; KOH, Joyce Hwee Ling. Review on teaching and learning of computational thinking through programming: What is next for K-12? **Computers in Human Behavior**, [S.l.], v. 41, p. 51–61, 2014. Disponível em: <https://linkinghub.elsevier.com/retrieve/pii/S0747563214004634>. Acesso em: 20 jun. 2019.
- MACHADO, João Luís de Almeida. **Alfabetização Digital: mais que um conceito, uma necessidade** - Educação - cmais+ O portal de conteúdo da Cultura. 2012. Disponível em: <http://culturafm.cmais.com.br/educacao/titulo-58>. Acesso em: 6 ago. 2018.
- MILLER, Jon D. Scientific Literacy: A conceptual and empirical review. **Daedalus**, [S.l.], v. 112, n. 2, p. 29–48, 1983. Disponível em: <https://www.jstor.org/stable/20024852>. Acesso em: 16 ago. 2018.
- MINAYO, Maria C. de Souza *et al.* **Pesquisa Social: teoria método e criatividade**. 21. ed. Petrópolis, RJ: Vozes, 2002.
- MINCHILLO, Laís Vasconcellos. **Na direção de melhores ferramentas e metodologias para o ensino de pensamento computacional para crianças**. 2018. Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Campinas, Campinas, 2018.
- MOHALLEM, D. F. *et al.* Avaliação do coeficiente de variação como medida da precisão em experimentos com frangos de corte. **Arquivo Brasileiro de Medicina Veterinária e Zootecnia**, [S.l.], v. 60, n. 2, p. 449–453, 2008.
- MORAES, Marcelo Botelho da Costa. **Análise multivariada aplicada à contabilidade**. São Paulo, 2016. Disponível em: https://edisciplinas.usp.br/pluginfile.php/1582605/mod_resource/content/1/AnáliseMulti-variada-Aula01.pdf. Acesso em: 1 abr. 2020.

MOREIRA, Marco Antonio. Aprendizagem significativa: da visão clássica à visão crítica. *In*: ENCONTRO INTERNACIONAL SOBRE APRENDIZAGEM SIGNIFICATIVA, V., ENCUESTRO NACIONAL SOBRE ENSEÑANZA DE LA MATEMÁTICA, I., 2006, Madrid, Espanha; Tandil, Argentina. **Anais [...]**. Madrid, Espanha; Tandil, Argentina. Disponível em: <http://www.if.ufrgs.br/~moreira/visaoclasicavisaocritica.pdf>. Acesso em: 20 dez. 2018.

MOREIRA, Marco Antonio. **Mapas conceituais e aprendizagem significativa**. 1. ed. São Paulo: Centauro, 2010.

MOREIRA, Marco Antonio. Aprendizagem significativa em mapas conceituais. **Textos de apoio ao Professor de Física**, Porto Alegre, v. 24, n. 6, p. 55, 2013. Disponível em: http://www.if.ufrgs.br/public/tapf/v24_n6_moreira_.pdf. Acesso em: 14 mar. 2020.

MOREIRA, Marco Antonio; MASINI, E. F. S. **Aprendizagem significativa: a teoria de David Ausubel**. 2. ed. São Paulo: Centauro, 2006.

MORENO, Julián. Digital competition game to improve programming skills. **Educational Technology and Society**, [S.l.], v. 15, n. 3, p. 288–297, 2012. Disponível em: <https://eric.ed.gov/?id=EJ992538>. Acesso em: 2 fev. 2019.

MORENO LEÓN, Jesús; ROBLES, Gregório; ROMÁN GONZÁLEZ, Marcos. Dr. Scratch : Análisis Automático de Proyectos Scratch para Evaluar y Fomentar el Pensamiento Computacional. **Revista de Educación a Distancia**, Murcia, Espanha, v. 46, n. 10, 2015. Disponível em: <https://revistas.um.es/red/article/view/240251>. Acesso em: 15 dez. 2018.

MORENO LEÓN, Jesús; ROMÁN GONZÁLEZ, Marcos; ROBLES, Gregório. On computational thinking as a universal skill: A review of the latest research on this ability. *In*: IEEE GLOBAL ENGINEERING EDUCATION CONFERENCE (EDUCON), 2018, Tenerife, Espanha. **Anais [...]**. Tenerife, Espanha: IEEE, 2018. Disponível em: <https://ieeexplore.ieee.org/abstract/document/8363437>. Acesso em: 22 jan. 2019.

MORETTO, Vasco P. Reflexões construtivistas sobre habilidades e competências. **Dois Pontos: Teoria & Prática em Gestão**, Belo Horizonte, MG, n. 42, p. 54–54, 1999.

MÜLLER, Luana. **Uma abordagem semiótica para apoiar programadores iniciantes durante o processo de reuso e de apropriação de códigos-fonte**. 2017. Tese (Doutorado em Ciência da Computação) - Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, 2017.

NAP. **Report of a Workshop on The Scope and Nature of Computational Thinking**. Washington, D.C.: National Academies Press, 2010. Disponível em: <http://www.nap.edu/catalog/12840>. Acesso em: 11 mar. 2019.

NCE. **National curriculum in England: computing programmes of study - GOV.UK**. 2013. Disponível em: <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study>. Acesso em: 25 mar. 2019.

NOVAK, Joseph Donald. **Uma teoria de Educação**. São Paulo: Biblioteca Pioneira, 1981.

NOVAK, Joseph Donald. Ayudar a los alumnos a aprender cómo aprender. La opinión de un profesor-investigador. **Enseñanza de las ciencias**, [S.l.], v. 9, n. 3, p. 215–228, 1983. Disponível em: <https://www.raco.cat/index.php/Ensenanza/article/view/39895>. Acesso em: 20 maio. 2019.

NOVAK, Joseph Donald. **Learning, creating, and using knowledge**. 2. ed. New York, USA: Taylor & Francis, 2010.

ONU. **The Age of Digital Interdependence** - Report of the UN Secretary-General's High-level Panel on Digital Cooperation. Nova York, 2019. Disponível em: <https://digitalcooperation.org/wp-content/uploads/2019/06/DigitalCooperation-report-for-web.pdf>. Acesso em: 15 nov. 2019.

OPMANIS, Martins; DAGIENE, Valentina; TRUU, Ahto. Task types at “beaver” contests. *In*: **INFORMATICS IN SECONDARY SCHOOLS: EVOLUTION AND PERSPECTIVES 2006**, Vilnius, Lithuania. **Anais [...]**. Vilnius, Lithuania: Institute of Mathematics and Informatics, 2006. Disponível em: <https://www.bebbras.org/sites/default/files/documents/publications/Opmanis-2006.pdf>. Acesso em: 2 abr. 2019.

PAPERT, Seymour. **LOGO: Computadores e Educação**. 1. ed. São Paulo: Editora Brasiliense, 1980a.

PAPERT, Seymour. **Mindstorms: children, computers and powerful ideas**. New York: Basic Books, 1980b.

PASQUAL JÚNIOR, Paulo Antonio. **Pensamento computacional e formação de professores: Uma análise a partir da plataforma Code.Org**. 2018. Dissertação (Mestrado em Educação) - Universidade de Caxias do Sul, Caxias do Sul, 2018.

PAZ, Louise Alessandra Santos do Carmo. O pensamento computacional e a formação continuada de professores: uma experiência com as TICs. **Revista on line de Política e Gestão Educacional**, Araraquara, SP, v. 21, n. esp.3, p. 1655–1667, 2017. Disponível em: <https://periodicos.fclar.unesp.br/rpge/article/view/10095>. Acesso em: 18 maio. 2019.

PELIZZARI, Adriana *et al.* Teoria da Aprendizagem Significativa segundo Ausubel. **PEC**, Curitiba, Paraná, v. 2, n. 1, p. 37–42, 2002. Disponível em: <http://portaldoprofessor.mec.gov.br/storage/materiais/0000012381.pdf>. Acesso em: 3 mar. 2020.

PEREIRA, Guilherme; ORTIGÃO, Maria Isabel Ramalho. Pesquisa Quantitativa em Educação: Algumas considerações. **Periferia**, Rio de Janeiro, v. 8, n. 1, p. 66–79, 2016. Disponível em: <https://www.e-publicacoes.uerj.br/index.php/periferia/article/viewFile/27341/19946>. Acesso em: 2 mar. 2020.

PERRENOUD, Philippe. **Construir as competências desde a Escola**. 1. ed. Porto Alegre: Artmed, 1999a.

PERRENOUD, Philippe. Construir competências é virar as costas aos saberes? **Pátio Revista Pedagógica**, Porto Alegre, n. 11, p. 15–19, 1999b. Disponível em:

http://www.unige.ch/fapse/SSE/teachers/perrenoud/php_main/php_1999/1999_39.html#Heading2. Acesso em: 15 maio. 2020.

PERRENOUD, Philippe *et al.* **As competências para ensinar no século XXI: a formação dos professores e o desafio da avaliação**. Porto Alegre: Artmed, 2002.

PRENSKY, Marc. **Programming is the New Literacy**. 2008. Disponível em: <https://www.edutopia.org/literacy-computer-programming>. Acesso em: 12 set. 2018.

RAABE, André *et al.* Um Instrumento para Diagnóstico do Pensamento Computacional. *In*: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (CBIE), VI., 2017, Recife. **Anais [...]**. Recife: SBC, 2017. Disponível em: <http://www.br-ie.org/pub/index.php/wcbie/article/view/7506>. Acesso em: 25 fev. 2019.

RAINIE, Lee; ANDERSON, Janna. **Code-Dependent: Pros and Cons of the Algorithm Age**. [S.l.] : Pew Research Center, 2017. Disponível em: <http://www.pewinternet.org/2017/02/08/code-dependent-pros-and-cons-of-the-algorithm-age>. Acesso em: 25 jun. 2019.

RAMOS, M. **Da qualificação à competência: deslocamento conceitual na relação trabalho-educação**. 2001. Tese (Doutorado em Educação) - Universidade Federal Fluminense, Niterói, 2001.

RAZALI, Nornadiah; WAH, Yap Bee. Power comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling tests. **Journal of Statistical Modeling and Analytics**, [S.l.], v. 2, n. 1, p. 13–14, 2011. Disponível em: https://www.researchgate.net/publication/267205556_Power_Comparisons_of_Shapiro-Wilk_Kolmogorov-Smirnov_Lilliefors_and_Anderson-Darling_Tests. Acesso em: 1 abr. 2020.

REHFELDT, Márcia Jussara Hepp. **A aplicação de Modelos Matemáticos em Situações-Problema Empresariais com uso do Software Lindo**. 2009. Tese (Doutorado em Informática na Educação) - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2009.

REIS, Marcelo Menezes. **Estatística aplicada à Administração**. Florianópolis, SC: Departamento de Ciências da Administração- UFSC, 2008.

RESNICK, Mitchel *et al.* Scratch Programming for All. **Communications of the ACM**, [S.l.], v. 52, n. 11, p. 60, 2009. Disponível em: <http://portal.acm.org/citation.cfm?doid=1592761.1592779>. Acesso em: 13 jan. 2019.

RESNICK, Mitchel. **Coding as the New Literacy - Serious Science**. 2015. Disponível em: <http://serious-science.org/coding-as-new-literacy-1546>. Acesso em: 13 set. 2018.

RIBEIRO, Leila; FOSS, Luciana; CAVALHEIRO, Simone André da Costa. **Entendendo o Pensamento Computacional**. [S.l.], 2017. Disponível em: <http://arxiv.org/abs/1707.00338>. Acesso em: 18 mar. 2019.

ROAZZI, Antonio; HODGES, Luciana. Compreensão de texto e modelos teóricos explicativos: a influência de fatores linguísticos, cognitivos e metacognitivos. *In*:

MOTTA, MÁRCIA P., SPINILLO, A. (ed.). **Compreensão de texto**. São Paulo: Casa Psicólogo, 2013.

ROCHA, Enilton Ferreira. Os dez pressupostos andragógicos da aprendizagem do adulto: um olhar diferenciado na educação do adulto. **Associação Brasileira de Educação a Distância**, [S.l.], 2012. Disponível em: http://www.abed.org.br/site/pt/midiateca/textos_ead/1102/http://www.abed.org.br/arquivos/os_10_pressupostos_andragogicos_eniltonpdf. Acesso em: 30 jul. 2019.

RODRIGUES, Rivanilson da Silva *et al.* Análise dos efeitos do Pensamento Computacional nas habilidades de estudantes no Ensino Básico: um estudo sob a perspectiva da programação de computadores. *In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (SBIE), XXVI .*, 2015, Maceió. **Anais [...]**. Maceió: SBC, 2015. Disponível em: <https://www.br-ie.org/pub/index.php/sbie/issue/view/129>. Acesso em: 4 fev. 2018.

ROMÁN GONZÁLEZ, Marcos. Aprender a programar “Apps” como enriquecimiento curricular en alumnado de alta capacidad. **Bordón Revista de Pedagogía**, Ciudad de México, México, v. 66, n. 4, p. 155, 2014. Disponível em: <https://recyt.fecyt.es/index.php/BORDON/article/view/28671/0>. Acesso em: 13 jan. 2019.

ROMÁN GONZÁLEZ, Marcos. **Codigoalfabetización y Pensamiento Computacional en Educación Primaria y Secundaria: Validación de un Instrumento y Evaluación de Programas**. 2016. Tese (Doctorado En Educación) - Universidad Nacional de Educación a Distancia, Madrid, 2016.

ROMÁN GONZÁLEZ, Marcos; PÉREZ GONZÁLEZ, Juan Carlos; JIMÉNEZ FERNÁNDEZ, Carmen. Test de Pensamiento Computacional: diseño y psicometría general. *In: CONGRESO INTERNACIONAL SOBRE APRENDIZAJE, INNOVACIÓN Y COMPETITIVIDAD (CINAIC), 3.*, 2015, Madrid. **Anais [...]**. Madrid: CINAIC, 2015. Disponível em: https://www.academia.edu/21309960/Test_de_Pensamiento_Computacional_diseño_y_psicometría_general_Computational_Thinking_Test_design_and_general_psychometry_. Acesso em: 10 maio. 2018.

ROMÁN GONZÁLEZ, Marcos; PÉREZ GONZÁLEZ, Juan Carlos; JIMÉNEZ FERNÁNDEZ, Carmen. Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. **Computers in human behavior**, [S.l.], v. 72, p. 678–691, 2017. Disponível em: <https://linkinghub.elsevier.com/retrieve/pii/S0747563216306185>. Acesso em: 4 abr. 2019.

ROSCOE, Jonathan Francis; FEARN, Stephen; POSEY, Emma. Teaching computational thinking by playing games and building robots. *In: INTERNATIONAL CONFERENCE ON INTERACTIVE TECHNOLOGIES AND GAMES (ITAG), 2014*, Nottingham, UK. **Anais [...]**. Nottingham, UK: IEEE, 2014. Disponível em: <https://ieeexplore.ieee.org/abstract/document/6990184>. Acesso em: 12 jun. 2018.

RUIPÉREZ, Beatriz Ortega. **Pensamiento Computacional y Resolución de Problemas**. 2017. Tese (Doctorado en Psicología) - Universidad Autónoma de Madrid,

Madrid, 2017. Disponível em: <https://repositorio.uam.es/handle/10486/683810>. Acesso em: 15 abr. 2019.

RUSHKOFF, Douglas. **Code Literacy: A 21st-Century Requirement**. 2012. Disponível em: <https://www.edutopia.org/blog/code-literacy-21st-century-requirement-douglas-rushkoff>. Acesso em: 15 ago. 2018.

SANTANA, Bianca Leite. **Uma abordagem de Ensino-Aprendizagem de Programação na Educação Superior**. 2018. Dissertação (Mestrado em Computação Aplicada) - Universidade Estadual de Feira de Santana, Feira de Santana, 2018.

SANTANNA, Hugo Cristo. **Ação, Computação, Representação** - Uma investigação psicogenética sobre o desenvolvimento do Pensamento Computacional. 2014. Tese (Doutorado em Psicologia) - Universidade Federal do Espírito Santo, Vitória, 2014.

SANTOS, César C. Andragogia: aprendendo a ensinar adultos. Simpósio de Excelência em Gestão e Tecnologia – 2010. Disponível em: http://www.aedb.br/seget/arquivos/artigos10/402_ArtigoAndragogia.pdf. Acesso em: 01 nov. 2018.

SANTOS, Wildson Luiz Pereira dos. Educação científica na perspectiva de letramento como prática social: funções, princípios e desafios. **Revista Brasileira de Educação**, São Paulo, v. 12, n. 36, p. 474–492, 2007. Disponível em: <https://www.scielo.br/pdf/rbedu/v12n36/a07v1236.pdf>. Acesso em: 31 jul. 2018.

SANTOS, José Ribamar A. **Gamificação no Ensino-Aprendizagem de Algorítmicos e Lógica Aplicada à Computação**. 2018. Dissertação (Mestrado em Ciência da Computação) - Faculdade Campo Limpo Paulista, Campo Limpo Paulista, 2018.

SASSERON, Lúcia Helena; CARVALHO, Anna Maria P. Investigações em ensino de ciências. **Investigações em Ensino de Ciências**, Porto Alegre, RS, v. 16, n. 1, p. 59–77, 2011. Disponível em: <https://www.if.ufrgs.br/cref/ojs/index.php/ienci/article/view/246/172>. Acesso em: 15 ago. 2018.

SEITER, Linda; FOREMAN, Brendan. Modeling the learning progressions of computational thinking of primary grade students. *In: ANNUAL INTERNATIONAL ACM CONFERENCE ON INTERNATIONAL COMPUTING EDUCATION RESEARCH (ICER)*, 9., 2013, New York, USA. **Anais [...]**. New York, USA: ACM Press, 2013. Disponível em: <http://dl.acm.org/citation.cfm?doid=2493394.2493403>. Acesso em: 3 abr. 2019.

SHERMAN, Mark; MARTIN, Fred. The assessment of mobile computational thinking. **Journal of Computing Sciences in Colleges**, [S.l.], v. 30, n. 6, p. 53–59, 2015. Disponível em: <http://www.cs.uml.edu/~fredm/papers/p53-sherman.pdf>. Acesso em: 12 abr. 2019.

SHIMAKURA, Sílvia. **Coefficiente de Variação**. 2005. Disponível em: <http://leg.ufpr.br/~silvia/CE701/node24.html>. Acesso em: 15 abr. 2020.

SILVA, André Luís Silva da; DEL PINO, José Claudio. **Metodologias de Ensino** - No contexto da formação continuada de professores. Curitiba, Paraná: Appris, 2019.

SILVA, Eliel Constantino da. **Pensamento computacional e a formação de conceitos matemáticos nos anos finais do Ensino Fundamental:** Uma possibilidade com kits de robótica. 2018. Dissertação (Mestrado em Educação Matemática) - Universidade Estadual Paulista “Júlio de Mesquita Filho”, Rio Claro, 2018.

SILVA, Tatyane Souza Calixto da. **Um modelo para promover o engajamento estudantil e auxiliar o aprendizado de programação utilizando gamification.** 2016. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal de Pernambuco, Recife, 2016.

SILVA JÚNIOR, Augusto Márcio da. **Microgênese do desenvolvimento sociocultural do raciocínio lógico-matemático mediado por tecnologias educacionais.** 2018. Dissertação (Mestrado em Educação) - Universidade Federal de Alfenas, Alfenas, MG, 2018.

SILVA, Paulo H. A.; LIMA, Alessandro S. **Cursos Binacionais:** relatos de uma experiência inovadora. Santana do Livramento: Cia do eBook, 2015.

SOARES, Magda. **Letramento : um tema em três gêneros.** Belo Horizonte: Autêntica, 1998.

SOSTER, Tatiana Sansone. **Revelando as essências da Educação Maker:** percepções das teorias e das práticas. 2018. Tese (Doutorado em Educação - Currículo) - Pontifícia Universidade Católica de São Paulo, São Paulo, 2018.

SOUZA, Isabelle Maria Lima de; RODRIGUES, Rivanilson da Silva; ANDRADE, Wilkerson. Introdução do pensamento computacional na formação docente para ensino de robótica educacional. *In: WORKSHOPS DO CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (CBIE)*, 5, 2016, Uberlândia. **Anais [...]**. Uberlândia: SBC, 2016. Disponível em: <http://br-ie.org/pub/index.php/wcbie/article/view/7052>. Acesso em: 11 abr. 2019.

STEVENSON, William J. **Estatística aplicada à Administração.** São Paulo: Harbra, 2001.

TAKAHASHI, Tadao. **Sociedade da Informação no Brasil Livro Verde.** Brasília. Disponível em: <http://www.socinfo.org.br>. Acesso em: 22 jul. 2020.

TANG, Xiaodan *et al.* Assessing computational thinking: A systematic review of empirical studies. **Computers and Education**, [S.l.], v. 148, p. 103798, 2020. Disponível em: <https://doi.org/10.1016/j.compedu.2019.103798>. Acesso em: 3 jun. 2020.

THE ROYAL SOCIETY. **Shut down or restart ?** Londres, 2012. Disponível em: <https://royalsociety.org/-/media/education/computing-in-schools/2012-01-12-computing-in-schools.pdf>. Acesso em: 23 dez. 2018.

TOWHIDNEJAD, M. *et al.* Introducing computational thinking through stealth teaching. *In: FRONTIERS IN EDUCATION CONFERENCE (FIE)*, 45., 2015, El Paso, Texas. **Anais [...]**. El Paso, Texas: IEEE, 2015. Disponível em: <http://ieeexplore.ieee.org/document/7044407/>. Acesso em: 5 abr. 2019.

TRIVIÑOS, Augusto N. S. **Introdução à pesquisa em Ciências Sociais - A pesquisa qualitativa em Educação**. São Paulo: Atlas, 2015.

TUCKER, Allen B.; NOOMAN, Robert. **Linguagens de Programação - Princípios e Paradigmas**. 2. ed. Porto Alegre: AMGH, 2010.

VALENTE, José A. Integração do pensamento computacional no currículo da educação básica: diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. **Revista e-Curriculum**, São Paulo, v. 14, n. 3, p. 864–897, 2016. Disponível em: <https://revistas.pucsp.br/index.php/curriculum/article/view/29051>. Acesso em: 6 jun. 2018.

VALVERDE BERROCOSO, Jesús. Estrategias educativas para el desarrollo de la competencia digital. In: SANDOVAL ROMERO, Y. *et al* (ed.). **Las tecnologías de la información en contextos educativos: nuevos escenarios de aprendizaje**. 1. ed. Santiago de Cali: Universidad Santiago de Cali, 2012. p. 55–68.

VAN DYNE, Michele; BRAUN, Jeffrey. Effectiveness of a computational thinking (CS0) course on student analytical skills. In: ACM TECHNICAL SYMPOSIUM ON COMPUTER SCIENCE EDUCATION - (SIGCSE), 45., 2014, New York, USA. **Anais [...]**. New York, USA: ACM Press, 2014. Disponível em: <http://dl.acm.org/citation.cfm?doid=2538862.2538956>. Acesso em: 28 mar. 2019.

VASCONCELOS, Vitor Vieira; FEITOSA, Flávia da Fonseca. **Correlação Estatística**. 2017. Disponível em: https://pt.slideshare.net/vitor_vasconcelos/correlao-64713398. Acesso em: 9 abr. 2020.

WENGER, E. **Cultivating communities of practice: a quick start-up guide for communities of practice**. 2002. Disponível em: <http://wenger-trayner.com/project/community-of%0A-practice-start-up-guide/>. Acesso em: 18 nov. 2018.

WICKHAM, Hadley; STRYJEWSKI, Lisa. 40 Years of Boxplots. **Had.Co.Nz**, [S.l.], p. 1–17, 2011. Disponível em: <http://had.co.nz/stat645/project-03/boxplots.pdf>. Acesso em: 5 abr. 2020.

WIEBE, Eric N. *et al*. Development of a lean computational thinking abilities assessment for middle grades students. In: ACM TECHNICAL SYMPOSIUM ON COMPUTER SCIENCE EDUCATION (SIGCSE), 50., 2019, Minneapolis. **Anais [...]**. Minneapolis: ACM Press, 2019. Disponível em: <https://dl.acm.org/citation.cfm?id=3287390>. Acesso em: 28 jun. 2019.

WILSON, Carolyne *et al*. **Alfabetización mediática e informacional : curriculum para profesores**. Paris: Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura, UNESCO, 2011. Disponível em: <https://unesdoc.unesco.org/ark:/48223/pf0000216099>. Acesso em: 13 jan. 2019.

WING, Jeannette M. Computational thinking. **Communications of the ACM**, [S.l.], v. 49, n. 3, p. 33–35, 2006. Disponível em: <http://portal.acm.org/citation.cfm?doid=1118178.1118215>. Acesso em: 18 mar. 2019.

WING, Jeannette M. Computational thinking and thinking about computing.

Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, [S.l.], v. 366, n. 1881, p. 3717–3725, 2008.

WING, Jeannette M. Computational thinking's influence on research and education for all. **Italian Journal of Educational Technology**, [S.l.], v. 25, n. 2, p. 7–14, 2017. Disponível em: <https://www.learntechlib.org/p/183466/>. Acesso em: 18 mar. 2019.

XAVIER, Antonio C. dos S. **Letramento Digital e Ensino**. Recife: Núcleo de Estudos de Hipertexto e Tecnologia Educacional, 2015. Disponível em: <http://nehte.com.br/artigos/Letramento-Digital-Xavier.pdf>. Acesso em: 2 ago. 2018.

YADAV, Aman *et al.* Computational Thinking in Elementary and Secondary Teacher Education. **ACM Transactions on Computing Education**, [S.l.], v. 14, n. 1, p. 1–16, 2014. Disponível em: <http://dl.acm.org/citation.cfm?doid=2600089.2576872>. Acesso em: 18 mar. 2019.

YADAV, Aman *et al.* **Sowing the seeds of assessment literacy in secondary computer science education: A landscape study**. 2015. Disponível em: <http://csta.acm.org/Research/sub/Projects/AssessmentStudy2015.html>. Acesso em: 19 set. 2019.

YIN, Roberto K. **Estudo de Caso: Planejamento e Métodos**. 2. ed. Porto Alegre: Bookman, 2001.

YTURRALDE, Ernesto. **Antropogogía: la educación permanente del ser humano**. 2018. Disponível em: <http://antropogogia.com/>. Acesso em: 30 maio. 2019.

APÊNDICES

APÊNDICE A - Teses e suas Classificações

Dados da Tese	AC⁷⁷	AV⁷⁸
AMARAL, ERICO MARCELO HOFF do. Processo de ensino e aprendizagem de algoritmos integrando ambientes imersivos e o paradigma de blocos de programação visual' 20/08/2015 255 f. Doutorado em INFORMÁTICA NA EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, Porto Alegre.	A1	AV1
ANNA, HUGO CRISTO SANT. Ação, computação, representação: Uma investigação psicogenética sobre o desenvolvimento do pensamento computacional' 30/05/2014. Doutorado em PSICOLOGIA Instituição de Ensino: UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO, Vitória.	A4	AV6
ARAUJO, ANA LIZ SOUTO OLIVEIRA DE. Quantifying Computational Thinking Abilities 29/08/2019 97 f. Doutorado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE CAMPINA GRANDE, Campina.	A2	AV2
BARCELOS, THIAGO SCHUMACHER. Relações entre o Pensamento Computacional e a Matemática em atividades didáticas de construção de jogos digitais 27/03/2014 278 f. Doutorado em ENSINO DE CIÊNCIAS Instituição de Ensino: UNIVERSIDADE CRUZEIRO DO SUL, São Paulo.	A3	AV3
BOUCINHA, RAFAEL MARIMON. Aprendizagem do Pensamento Computacional e Desenvolvimento do Raciocínio 21/08/2017 151 f. Doutorado em INFORMÁTICA NA EDUCAÇÃO Instituição de Ensino:	A1	AV1

⁷⁷ AC: Área de Conhecimento

⁷⁸ AV: Área de Avaliação

Dados da Tese	AC⁷⁷	AV⁷⁸
UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, Porto Alegre.		
BRACKMANN, CHRISTIAN PUHLMANN. Desenvolvimento do Pensamento Computacional através de Atividades Desplugadas na Educação Básica 22/08/2017 226 f. Doutorado em INFORMÁTICA NA EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, Porto Alegre.	A1	AV1
CASTILHO, MARIA INES. Hiperobjetos da Robótica Educacional como Ferramentas para o Desenvolvimento da Abstração Reflexionante e do Pensamento Computacional 16/10/2018 214 f. Doutorado em INFORMÁTICA NA EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, Porto Alegre.	A1	AV1
FROTA, VITOR BREMGARTNER da. Arcabouço Conceitual de Adaptação de Recursos Educacionais' 24/11/2017 219 f. Doutorado em INFORMÁTICA Instituição de Ensino: UNIVERSIDADE FEDERAL DO AMAZONAS, Manaus.	A2	AV2
MORAIS, ANUAR DAIAN DE. O Desenvolvimento do Raciocínio Condicional a partir do Uso de Teste no Squeak Etoys 14/07/2016 <i>undefined</i> f. Doutorado em INFORMÁTICA NA EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL, Porto Alegre.	A1	AV1
MULLER, LUANA. Uma Abordagem Semiótica para Apoiar Programadores Iniciantes Durante o Processo de Reúso e de Apropriação de Códigos-Fonte 08/12/2017 108 f. Doutorado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: PONTIFÍCIA	A2	AV2

Dados da Tese	AC⁷⁷	AV⁷⁸
UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL, Porto Alegre.		
Nunes, João Fernando Igansi. Design computacional: comunicação do in-visível 01/05/2009 211 f. Doutorado em COMUNICAÇÃO E SEMIÓTICA Instituição de Ensino: PONTIFÍCIA UNIVERSIDADE CATÓLICA DE SÃO PAULO, SÃO PAULO Biblioteca Depositária: Biblioteca da PUC/SP Trabalho anterior à Plataforma Sucupira	A5	AV5
OLIVEIRA, ANDRE ROSA DE. Metadados Como Atributos Da Informação Estruturada Em Bases De Dados Jornalísticas Na Web 19/09/2016 163 f. Doutorado em COMUNICAÇÃO SOCIAL Instituição de Ensino: UNIVERSIDADE METODISTA DE SÃO PAULO, São Bernardo do Campo.	A5	AV5
SANTIN, MATEUS MADAIL. Desenvolvimento do pensamento computacional através da robótica: Fluidez digital no ensino fundamental 24/10/2014 134 f. Doutorado em EDUCAÇÃO EM CIÊNCIAS QUÍMICA DA VIDA E SAÚDE (UFSM - FURG) Instituição de Ensino: UNIVERSIDADE FEDERAL DO RIO GRANDE, Porto Alegre.	A3	AV3
SANTOS, CLOGOGIL FABIANO RIBEIRO DOS. A Robótica Educacional Como Recurso De Mobilização E Explicitação De Invariantes Operatórios Na Resolução De Problemas 29/11/2018 206 f. Doutorado em ENSINO DE CIÊNCIA E TECNOLOGIA Instituição de Ensino: UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ, Ponta Grossa.	A3	AV3
SOSTER, TATIANA SANSONE. Revelando as essências da educação maker: percepções das	A5	AV4

Dados da Tese	AC⁷⁷	AV⁷⁸
teorias e das práticas 16/08/2018 175 f. Doutorado em EDUCAÇÃO (CURRÍCULO) Instituição de Ensino: PONTIFÍCIA UNIVERSIDADE CATÓLICA DE SÃO PAULO, São Paulo.		

Legenda:

	Área Conhecimento
A1	Sociais e Humanidades
A2	Ciência da Computação
A3	Ensino
A4	Psicologia
A5	Comunicação
A6	Educação

	Área Avaliação
AV1	Interdisciplinar
AV2	Ciência da Computação
AV3	Ensino
AV4	Educação
AV5	Comunicação e Informação
AV6	Psicologia

Apêndice B – Dissertações e suas Classificações

<i>Informação da Dissertação</i>	<i>AC</i> ⁷⁹	<i>AV</i> ⁸⁰	<i>Nível</i>
ACHUTTI, CAMILA FERNANDEZ. Tree Bark framework: competences and mindset rearrangements for Digital and Technology Literacy in times of exponential rate of changes. 09/06/2017 221 f. Mestrado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE DE SÃO PAULO, São Paulo.	A1	AV1	O
ALVES, NATHALIA DA CRUZ. Codemaster: Um Modelo De Avaliação Do Pensamento Computacional Na Educação Básica Através Da Análise De Código De Linguagem De Programação Visual 13/02/2019 143 f. Mestrado em CIÊNCIAS DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE SANTA CATARINA, Florianópolis.	A1	AV1	EF
BALATON, MARIANA CARDOSO. Robótica Educacional Livre: um relato de prática no Ensino Fundamental 01/02/2017 108 f. Mestrado em EDUCAÇÃO (CURRÍCULO) Instituição de Ensino: PONTIFÍCIA UNIVERSIDADE CATÓLICA DE SÃO PAULO, São Paulo.	A2	AV2	EF
BARBOSA, LARA MARTINS. Aspectos do pensamento computacional na construção de fractais com o software GeoGebra 11/01/2019 168 f. Mestrado em EDUCAÇÃO MATEMÁTICA Instituição de Ensino: UNIVERSIDADE ESTADUAL PAULISTA	A8	AV3	ES

⁷⁹ AC: Área de Conhecimento

⁸⁰ AV: Área de Avaliação

<i>Informação da Dissertação</i>	<i>AC⁷⁹</i>	<i>AV⁸⁰</i>	<i>Nível</i>
JÚLIO DE MESQUITA FILHO (RIO CLARO), Rio Claro.			
BARCAROLI, VELCIR. Plataforma interativa de aprendizagem de programação voltada a disseminação do pensamento computacional utilizando robótica remota 19/06/2017. Mestrado Profissional em Computação Aplicada Instituição de Ensino: FUNDAÇÃO UNIVERSIDADE DE PASSO FUNDO, Passo Fundo.	A1	AV1	O
BENARROS, CYNARA RODRIGUES. Sequência Didática Para o Ensino-Aprendizagem de Informática no Curso de Assistente Administrativo do Ensino Profissional 30/03/2017 96 f. Mestrado Profissional em Ensino Tecnológico Instituição de Ensino: INSTITUTO FEDERAL DE EDUC., CIÊNCIA E TECNOLOGIA DO AMAZONAS, Manaus Biblioteca Depositária: Instituto Federal de Educação, Ciência e Tecnologia do Amazonas.	A3	AV3	O
BOMBASAR, JAMES ROBERTO. Computability Game - Um jogo de lógica inspirado na máquina de turing para apoio ao desenvolvimento do pensamento computacional 23/02/2017 112 f. Mestrado em COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.	A1	AV1	EB
BOZOLAN, SANDRA MUNIZ. O pensamento computacional: ensino e aprendizagem através do software processing 16/11/2016 145 f. Mestrado em TECNOLOGIAS DA INTELIGÊNCIA E DESIGN DIGITAL Instituição de Ensino: PONTIFÍCIA UNIVERSIDADE CATÓLICA DE SÃO PAULO, São Paulo.	A4	AV4	ES

<i>Informação da Dissertação</i>	<i>AC⁷⁹</i>	<i>AV⁸⁰</i>	<i>Nível</i>
BREMM, CRISTIANE INES. Mediação Do Pensamento Computacional E Programação No Processo De Interação Das Crianças Na Educação Infantil 13/07/2018. Mestrado Profissional em Tecnologias Educacionais em Rede Instituição de Ensino: UNIVERSIDADE FEDERAL DE SANTA MARIA, Santa Maria.	A6	AV4	EI
BRESSAN, MANUELLE LOPES QUINTAS. SCRATCH! Um Estudo De Caso 09/12/2016. Mestrado em TECNOLOGIA E SOCIEDADE Instituição de Ensino: UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ, Curitiba.	A6	AV4	EM/E F
BROCHADO, ELIANA ALICE. Scratch Como Fator Mobilizador Para Produção De Narrativas Digitais 04/12/2018 197 f. Mestrado em Educação Instituição de Ensino: UNIVERSIDADE FEDERAL DE ALFENAS, Alfenas Biblioteca Depositária: Biblioteca Central UNIFAL-MG.	A2	AV2	EF
CARBAJAL, MARLENY LUQUE. Design E Desenvolvimento de Um Ambiente De Programação Tangível De Baixo Custo Para Crianças 08/08/2016 86 f. Mestrado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE ESTADUAL DE CAMPINAS, Campinas.	A1	AV1	EF
CARNEIRO, ANA CAROLINA ROCHA. Um Estudo De Caso Sobre Informática Na Educação Infantil: A Transformação De Caça-Níqueis Em Computadores No Município De Balneário Camboriú, Santa Catarina 26/02/2014 226 f. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.	A2	AV2	EF

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
CARVALHO, FELIPE JOSE REZENDE DE. Introdução à programação de computadores por meio de uma tarefa de modelagem matemática na educação matemática 15/06/2018 133 f. Mestrado em ENSINO Instituição de Ensino: UNIVERSIDADE ESTADUAL DO OESTE DO PARANA, Foz do Iguaçu.	A3	AV3	EM
COSTA, ERICK JOHN FIDELIS. Pensamento Computacional Na Educação Básica: Uma Abordagem Para Estimular A Capacidade De Resolução De Problemas Na Matemática 02/03/2017 157 f. Mestrado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE CAMPINA GRANDE, Campina Grande.	A1	AV1	EB
COUTO, CARLA MACHADO. “Disciplina Informática” Na Educação Fundamental A Partir De Seus Professores 11/12/2018 107 f. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE ESTÁCIO DE SÁ, Rio de Janeiro.	A2	AV2	EF
COUTO, GABRIEL MILITELLO. Pensamento Computacional Educacional: Ensaio Sobre Uma Perspectiva Libertadora 07/08/2017 181 f. Mestrado em EDUCAÇÃO (CURRÍCULO) Instituição de Ensino: PONTIFÍCIA UNIVERSIDADE CATÓLICA DE SÃO PAULO, São Paulo.	A2	AV2	O
COUTO, KATIANE CUGIK. O Ensino De Programação Nos Anos Iniciais Do Ensino Fundamental: Do Estudo Do Pensamento Computacional À Proposta De Mídias Educacionais 28/06/2018 131 f. Mestrado Profissional em ENSINO DE CIÊNCIAS, MATEMÁTICA E TECNOLOGIAS Instituição de	A3	AV3	EF

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
Ensino: UNIVERSIDADE DO ESTADO DE SANTA CATARINA.			
COUTO, NATALIA ELLERY RIBEIRO. Impacto Da Implantação De Uma Disciplina De Pensamento Computacional Nos Estudantes: Um Estudo De Caso 30/07/2018 135 f. Mestrado em COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.	A1	AV1	ES
CROVADOR, ALVARO. O Uso Do Arduíno Em Sala De Aula No Experimento Do Pêndulo Simples De Galileu Galilei 19/03/2019 84 f. Mestrado Profissional em Educação e Novas Tecnologias Instituição de Ensino: CENTRO UNIVERSITÁRIO INTERNACIONAL, Curitiba.	A2	AV2	ES
CURASMA, HERMINIO PAUCAR. Uma ferramenta de realidade virtual para a introdução à programação e pensamento computacional para jovens 24/10/2017. Mestrado em INFORMÁTICA Instituição de Ensino: PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO, Rio de Janeiro.	A1	AV1	EF/ EM
EGIDO, SIDNEIA VALERO. Educação Matemática E Desenvolvimento Do Pensamento Computacional No 3º Ano Do Ensino Fundamental: Crianças Programando Jogos Com Scratch 23/03/2018 138 f. Mestrado em Educação em Ciências e em Matemática Instituição de Ensino: UNIVERSIDADE FEDERAL DO PARANÁ, Curitiba.	A8	AV3	EF
ELOY, ADELMO ANTONIO DA SILVA. Contribuições para aplicação de learning analytics no apoio à avaliação em atividades de introdução à programação com Scratch 07/08/2019 165 f.	A7	AV8	⁸¹

⁸¹ Arquivo não disponível

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
Mestrado em ENGENHARIA ELÉTRICA Instituição de Ensino: UNIVERSIDADE DE SÃO PAULO, São Paulo.			
EVARISTO, INGRID SANTELLA. O Pensamento Computacional No Processo De Aprendizagem Da Matemática Nos Anos Finais Do Ensino Fundamental 05/12/2019 173 f. Mestrado Profissional em Gestão e Práticas Educacionais Instituição de Ensino: UNIVERSIDADE NOVE DE JULHO, São Paulo.	A2	AV2	EF
FELIX, DOUGLAS FURTADO. O Pensamento Computacional No Ensino Fundamental: Um Estudo De Caso Baseado Nas Neurociências 19/10/2018 85 f. Mestrado em MODELAGEM COMPUTACIONAL Instituição de Ensino: UNIVERSIDADE FEDERAL DO RIO GRANDE, Rio Grande.	A4	AV4	EF
FERNANDES, HUGO BATISTA. Pensamento Computacional: Uma Proposta De Curso De Extensão On-Line Para Professores Que Lecionam Matemática Nos Anos Iniciais Do Ensino Fundamental 09/03/2018. Mestrado em ENSINO DE CIÊNCIAS Instituição de Ensino: UNIVERSIDADE CRUZEIRO DO SUL, São Paulo.	A3	AV3	ES
FERRI, JULIANA. Ensino de Linguagem de Programação na Educação Básica: uma proposta de sequência didática para desenvolver o pensamento computacional 28/09/2017 119 f. Mestrado Profissional em ENSINO Instituição de Ensino: UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ, Cornélio Procópio.	A3	AV3	EB
FORQUESATO, LUIS EDUARDO THIBES. Using a Game to Teach Computational Thinking and	A1	AV1	O

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
Assess Learning 20/12/2018 56 f. Mestrado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE ESTADUAL DE CAMPINAS, Campinas.			
FRANCA, ROZELMA SOARES DE. Um modelo para a aprendizagem do pensamento computacional aliado à autorregulação 23/02/2015. Mestrado em CIÊNCIAS DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE PERNAMBUCO, Recife.	A1	AV1	EM
GERALDES, WENDELL BENTO. O Pensamento Computacional No Ensino Profissional E Tecnológico 23/02/2017 81 f. Mestrado Profissional em Governança, Tecnologia e Inovação Instituição de Ensino: UNIVERSIDADE CATÓLICA DE BRASÍLIA, Brasília Biblioteca Depositária: Universidade Católica de Brasília.	A4	AV4	O
GOMES, TANCICLEIDE CARINA SIMOES. Desenvolvimento Do Pensamento Computacional Na Educação Infantil: Contribuições De Uma Pesquisa-Ação Educacional 21/02/2018 102 f. Mestrado em CIÊNCIAS DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE PERNAMBUCO, Recife.	A1	AV1	EI
GONCALVES, FILIPE AUGUSTO. Um instrumento para diagnóstico do pensamento computacional 17/12/2015 120 f. Mestrado em COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.	A1	AV1	ES
JENICHEN, NATALIA MUELLER. Os “Padrões De Competência Em TIC Para Professores” Estabelecidos Pela Unesco: Investigando O	A2	AV2	ES

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
Desejável E O Provável Na Percepção De Docentes Do Ensino Superior De Santa Catarina Para O Decênio 2014-2024 31/07/2014 116 f. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.			
JUNIOR, AUGUSTO MARCIO DA SILVA. Microgênese Do Desenvolvimento Sociocultural Do Raciocínio Lógico-Matemático Mediado Por Tecnologias Educacionais 26/02/2018 135 f. Mestrado em Educação Instituição de Ensino: UNIVERSIDADE FEDERAL DE ALFENAS, Alfenas.	A2	AV2	ES
JUNIOR, JOSE ETIENE BEZERRA. Investigando o Uso do Extreme Programming como uma Metodologia de Ensino para Aplicações Práticas da Robótica Educacional 29/03/2018 124 f. Mestrado em CIÊNCIA DA COMPUTAÇÃO - UERN - UFERSA Instituição de Ensino: UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE, Mossoró.	A1	AV1	EM
JUNIOR, PAULO ANTONIO PASQUAL. Pensamento Computacional E Formação De Professores: Uma Análise A Partir Da Plataforma Code.Org 03/07/2018 119 f. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE DE CAXIAS DO SUL, Caxias do Sul.	A2	AV2	ES
JUNIOR, VALDIR JOSE CORREA. Uma Experiência De Uso Do Geogebra Na Identificação De Padrões Em Trigonometria 17/12/2014 130 f. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.	A2	AV2	EF
LIMA, WILLIAM VIEIRA DE. Percepção Ambiental E Desenvolvimento Scratch: Uso Da Água No Pulsar Do Rio Juruá – Eirunepé – Amazonas 09/08/2018	A9	AV7	EM

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
115 f. Mestrado Profissional em Rede Nacional para Ensino das Ciências Ambientais Instituição de Ensino: UNIVERSIDADE FEDERAL DO AMAZONAS, São Carlos.			
LUMMERTZ, RAMON DOS SANTOS. As Potencialidades Do Uso Do Software Scratch Para A Construção Da Literacia Digital 17/05/2016 133 f. Mestrado em ENSINO DE CIÊNCIAS E MATEMÁTICA Instituição de Ensino: UNIVERSIDADE LUTERANA DO BRASIL, Canoas.	A8	AV3	EF
MATTOS, FRANCIELLE DE. Uma Iniciativa para Estimular o uso de Tecnologias por Meninas no Ensino Médio 05/11/2018 162 f. Mestrado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE SÃO CARLOS, São Carlos.	A1	AV1	EM
MEIRA, RICARDO RADAELLI. Pensamento Computacional Na Educação Básica: Uma Proposta Metodológica Com Jogos E Atividades Lúdicas 20/10/2017 119 f. Mestrado Profissional em Tecnologias Educacionais em Rede Instituição de Ensino: UNIVERSIDADE FEDERAL DE SANTA MARIA, Santa Maria.	A6	AV4	EF
MESTRE, PALLOMA ALENCAR ALVES. O Uso do Pensamento Computacional como Estratégia para Resolução de Problemas Matemáticos 08/03/2017 103 f. Mestrado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE CAMPINA GRANDE, Campina.	A1	AV1	O
MIGUEL, CAROLINA COSTA. O Papel Das Interações E Linguagens No Ensino De Ciências Tecnológicas No Contexto Da Educação Infantil	A2	AV2	EI

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
21/02/2019. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE SÃO CARLOS, São Carlos.			
MINCHILLO, LAIS VASCONCELLOS. Towards better tools and methodologies to teach computational thinking to children 13/06/2018. Mestrado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE ESTADUAL DE CAMPINAS, Campinas.	A1	AV1	EF
MORAIS, DYEGO CARLOS SALES DE. Modelo de Desenvolvimento Participativo de Jogos Digitais Educacionais no Contexto Escolar 26/08/2016 126 f. Mestrado em CIÊNCIAS DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE PERNAMBUCO, Recife.	A1	AV1	EM
NASCIMENTO, RENATA MELO. A Matemática E O Visualg: Lógica De Programação No Ensino Médio 09/09/2019 124 f. Mestrado Profissional em Matemática em Rede Nacional Instituição de Ensino: UNIVERSIDADE ESTADUAL DE SANTA CRUZ, Rio de Janeiro.	A5	AV5	EM
NUNES, CINTHIA BATISTA. Introdução à computação: uma proposta para o ensino básico 13/08/2013. Mestrado Profissional em Matemática em Rede Nacional Instituição de Ensino: UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA, Rio de Janeiro.	A5	AV5	EB
PAZINI, ERNANI ZANDONA. Arquitetura Paramétrica: mensuração do fenômeno de engajamento no processo de projeto contemporâneo 19/12/2018 194 f. Mestrado em	A10	AV6	ES

<i>Informação da Dissertação</i>	<i>AC⁷⁹</i>	<i>AV⁸⁰</i>	<i>Nível</i>
ARQUITETURA E URBANISMO Instituição de Ensino: FACULDADE MERIDIONAL, Passo Fundo.			
PEREIRA, JOAO PEDRO DE LIMA. Programação e Pensamento Computacional no 8° e 9° ano do Ensino Fundamental: Um Estudo de Caso 08/08/2019. Mestrado Profissional em Matemática em Rede Nacional Instituição de Ensino: UNIVERSIDADE DE BRASÍLIA, Rio de Janeiro.	A5	AV5	EF
POLONI, LEONARDO. Aprendizagem De Programação Mediada Por Uma Linguagem Visual: Possibilidade De Desenvolvimento Do Pensamento Computacional 16/04/2018 180 f. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE DE CAXIAS DO SUL, Caxias do Sul.	A2	AV2	EM
QUEIROZ, RUBENS LACERDA. DUINOBLOCKS4KIDS: utilizando tecnologia livre e materiais de baixo custo para o exercício do Pensamento Computacional no Ensino Fundamental I por meio do aprendizado de programação aliado à Robótica Educacional. 30/03/2017 243 f. Mestrado em INFORMÁTICA Instituição de Ensino: UNIVERSIDADE FEDERAL DO RIO DE JANEIRO, Rio de Janeiro.	A1	AV1	EF
RIBEIRO, MIRIAN REGINA PEREIRA. Mídia & Educação: Análise Da Percepção De Um Grupo De Estudantes Acerca De Ações Educativas Voltadas À Convivência Com As Mídias Digitais 28/07/2014 124 f. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.	A2	AV2	EM
RIBOLDI, SANDRA MARA OSELAME. A Linguagem De Programação Scratch E O Ensino De Funções: Uma Possibilidade 30/09/2019 108 f. Mestrado	A5	AV5	EF

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
Profissional em Matemática em Rede Nacional Instituição de Ensino: UNIVERSIDADE FEDERAL DA FRONTEIRA SUL, Rio de Janeiro.			
ROCHA, JOSE RAFAEL MORAES GARCIA DA. Developing Programming Skills On Digital Native Children Through The Interaction With Smart Devices 15/01/2016 77 f. Mestrado Profissional em CIÊNCIAS DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE PERNAMBUCO, Recife.	A1	AV1	EI
RODRIGUES, RIVANILSON DA SILVA. Um Estudo Sobre Os Efeitos Do Pensamento Computacional Na Educação 09/03/2017 95 f. Mestrado em CIÊNCIA DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE CAMPINA GRANDE, Campina.	A1	AV1	EB
ROSARIO, TATIANE APARECIDA MARTINS DO. As Aprendizagens Com O Uso Do Brinquedo De Programar: Um Estudo Com Crianças De Cinco E Seis Anos De Idade De Uma Instituição De Educação Infantil 24/08/2017 94 f. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.	A2	AV2	EI
ROUCAS, MARCIA DE FATIMA DUARTE. Scratch Como Ferramenta Pedagógica Para Estimular O Processo De Letramento Na Educação De Jovens E Adultos 17/12/2018 161 f. Mestrado Profissional em Diversidade e Inclusão Instituição de Ensino: UNIVERSIDADE FEDERAL FLUMINENSE, Niterói.	A3	AV3	O
SANTANA, ANDRE LUIZ MACIEL. Análise do processo metodológico de montagem de um brinquedo de programar 16/12/2015 93 f. Mestrado	A1	AV1	EM

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
em COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.			
SANTANA, BIANCA LEITE. Uma Abordagem de Ensino-Aprendizagem de Programação na Educação Superior 06/04/2018 164 f. Mestrado em Computação Aplicada Instituição de Ensino: UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA, Feira de Santana.	A4	AV4	ES
SANTOS, JOSE RIBAMAR AZEVEDO DOS. Gamificação No Ensino-Aprendizagem De Algorítmicos E Lógica Aplicada A Computação 31/01/2018 173 f. Mestrado em Ciência da Computação Instituição de Ensino: CENTRO UNIVERSITÁRIO CAMPO LIMPO PAULISTA, Campo Limpo Paulista.	A1	AV1	ES
SILVA, ELIEL CONSTANTINO DA. Pensamento computacional e a formação de conceitos matemáticos nos anos finais do Ensino Fundamental: uma possibilidade com kits de robótica 21/12/2018 264 f. Mestrado em EDUCAÇÃO MATEMÁTICA Instituição de Ensino: UNIVERSIDADE ESTADUAL PAULISTA JÚLIO DE MESQUITA FILHO (RIO CLARO), Rio Claro.	A8	AV3	EF
SILVA, FIRMIANO ALEXANDRE DOS REIS. Cartas à Mesa: Uma Proposta Lúdico-Didática Para O Ensino De Lógica 06/09/2019 92 f. Mestrado Profissional em EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA Instituição de Ensino: INSTITUTO FED. DE EDUC., CIÊNC. E TECN. DO TRIÂNGULO MINEIRO, Vitória.	A3	AV3	EM
SILVA, GLADYS SORAIA. Mediação Da Aprendizagem Com Uso De Tecnologia: Um	A2	AV2	EF

<i>Informação da Dissertação</i>	<i>AC⁷⁹</i>	<i>AV⁸⁰</i>	<i>Nível</i>
Estudo A Partir Da Modificabilidade Cognitiva 26/08/2014 132 f. Mestrado em EDUCAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.			
SILVA, HELIO MOREIRA DA. Pensamento Computacional: Desenvolvimento Do Material Pedagógico Para O Ensino De Programação De Games 2D Na Educação Básica Auxiliado Pelo Design De Interação 14/08/2019 71 f. Mestrado Profissional em DESIGN, TECNOLOGIA E INOVAÇÃO Instituição de Ensino: CENTRO UNIVERSITÁRIO TERESA D'ÁVILA, Lorena.	A11	AV6	EF
SILVA, JESSICA LAISA DIAS DA. Game Design de Jogos Digitais de Pensamento Computacional inspirados no Instrumento de Avaliação Bebras Challenge 25/07/2018. Mestrado em SISTEMAS E COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE, Natal.	A1	AV1	O
SILVA, MOISA APARECIDA DA. Criatividade Literária Na Autoria De Narrativas Digitais Multidisciplinares No Scratch 21/03/2019 172 f. Mestrado em Educação Instituição de Ensino: UNIVERSIDADE FEDERAL DE ALFENAS, Alfenas.	A2	AV2	EF
SILVA, TATYANE SOUZA CALIXTO DA. Um modelo para promover o engajamento estudantil e auxiliar o aprendizado de programação utilizando gamification 29/02/2016 154 f. Mestrado em CIÊNCIAS DA COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE FEDERAL DE PERNAMBUCO, Recife.	A1	AV1	ES

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
SOUZA, LEANDRO DELGADO DE. Instituto De Hackers: O Pensamento Computacional Aplicado Ao Ensino Técnico Integrado Ao Ensino Médio 07/10/2019 95 f. Mestrado Profissional em EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA Instituição de Ensino: INSTITUTO FEDERAL DE EDUC., CIÊNCIA E TECNOLOGIA DO PARANÁ, Vitória.	A3	AV3	EM
STELLA, ANA LUCIA. Utilizando O Pensamento Computacional E A Computação Criativa No Ensino Da Linguagem De Programação Scratch Para Alunos Do Ensino Fundamental 23/02/2016. Mestrado em Tecnologia Instituição de Ensino: UNIVERSIDADE ESTADUAL DE CAMPINAS (LIMEIRA), Limeira.	A4	AV4	EF
TASSANO, DEBORA PAOLA RODRIGUEZ. Um Olhar Sobre Teorias Cognitivas: Promovendo O Aprendizado De Lógica E Programação 14/03/2016 109 f. Mestrado Profissional em Educação e Tecnologia Instituição de Ensino: INSTITUTO FEDERAL DE EDUC., CIÊNC. E TECN. SUL-RIO-GRANDENSE, Pelotas.	A2	AV2	O
ZANCHETT, GUILHERME ALEXANDRE. Framework para auxiliar o desenvolvimento de jogos que abordem o pensamento computacional 15/12/2016 131 f. Mestrado em COMPUTAÇÃO Instituição de Ensino: UNIVERSIDADE DO VALE DO ITAJAÍ, Itajaí.	A1	AV1	O
ZIMMERMANN, JUSSARA SIQUEIRA DE OLIVEIRA. Aplicação E Avaliação de Conceitos Do Pensamento Computacional Em Pacientes De Um Hospital Pediátrico 09/11/2018 111 f. Mestrado em GESTÃO E INFORMÁTICA EM SAÚDE Instituição de	A12	AV4	O

Informação da Dissertação	AC⁷⁹	AV⁸⁰	Nível
Ensino: UNIVERSIDADE FEDERAL DE SÃO PAULO, São Paulo.			

Legenda:

	Área de Conhecimento
A1	Ciência da Computação
A2	Educação
A3	Ensino
A4	Eng./Tecnologia/Gestão
A5	Matemática
A6	Socias e Humanidades
A7	Eng. Elétrica
A8	Ensino de Ciências e Matemática
A9	Ciências Ambientais
A10	Arquitetura e Urbanismo
A11	Desenho Industrial
A12	Saúde e Biológicas

	Área de Avaliação
AV1	Ciência da Computação
AV2	Educação
AV3	Ensino
AV4	Interdisciplinar
AV5	Matemática/Probabilidade
AV6	Arquitetura, Urbanismo e Design
AV7	Ciências Ambientais
AV8	Engenharia IV

	Nível
EB	Educação Básica
EF	Ensino Fundamental
EI	Educação Infantil
EM	Ensino Médio
ES	Ensino Superior
O	Outro (não identificado)

APÊNDICE C - Termo de Consentimento Livre e Esclarecido (TCLE)

TERMO DE CONSENTIMENTO LIVRE E ESCLARECIDO (TCLE)

Prezado participante,

Você está sendo convidado(a) a participar da pesquisa "Desenvolvendo o Pensamento Computacional através da Programação de APP", desenvolvida por Walkiria Cordenonzi, aluna de Doutorado em Ensino da Universidade do Vale do Taquari - Univates, sob orientação do Professor Dr. José del Pino.

Sobre o objetivo central

O objetivo central do estudo é: analisar o desenvolvimento do PC nos sujeitos (já completaram o pós-médio) com formação em escolas brasileiras e uruguaias.

Por que o participante está sendo convidado (critério de inclusão)

O convite a sua participação se deve à sua matrícula em um curso que pertence ao acordo binacional firmado entre a *Consejo de Educacion Tecnico Profesional - Universidad del Trabajo del Uruguay* - CETP-UTU e o Instituto Federal de Ciência e Tecnologia Sul-riograndense - IFSUL.

Sua participação é voluntária, isto é, ela não é obrigatória, e você tem plena autonomia para decidir se quer ou não participar, bem como retirar sua participação a qualquer momento. Você não será penalizado de nenhuma maneira caso decida não consentir sua participação, ou desistir da mesma. Contudo, ela é muito importante para a execução da pesquisa.

Como pesquisadora, comprometo-me a esclarecer devida e adequadamente qualquer dúvida que, eventualmente, o/a participante venha a ter, no momento da pesquisa ou posteriormente. Serão garantidas a confidencialidade e a privacidade das informações por você prestadas.

Mecanismos para garantir a confidencialidade e a privacidade

Qualquer dado que possa identificá-lo será omitido na divulgação dos resultados da pesquisa, e o material será armazenado em local seguro.

A qualquer momento, durante a pesquisa, ou posteriormente, você poderá solicitar do pesquisador informações sobre sua participação e/ou sobre a pesquisa, o que poderá ser feito através dos meios de contato explicitados neste Termo.

Procedimentos detalhados que serão utilizados na pesquisa



Você participará desta pesquisa, que consiste em assistir o curso "Eu Programa 1.0!". Sua intervenção será como aluno, serão solicitados preenchimento de questionários, avaliações e construção de programas (aplicativos). Será utilizado um ambiente virtual de aprendizagem (AVA), no qual serão disponibilizados os documentos já citados e o material instrucional para uso durante o curso. Durante o curso, você receberá o atendimento necessário, tanto presencial quanto virtual, e para seu melhor aproveitamento, para todas as atividades realizadas, terão feedback individual.

Tempo de duração do procedimento

O tempo de duração: 20 horas distribuídas em 3 horas semanais, de acordo com a disponibilidade da Instituição.

Guarda dos dados e material coletados na pesquisa

Os documentos gerados nesta pesquisa ficarão em posse da pesquisadora, em formato digital. Sendo que somente a mesma e seu orientador terão o acesso. Ao final da pesquisa, todo material será mantido em arquivo, por pelo menos 5 anos, conforme Resolução CNS nº 466/12.

Explicitar benefícios diretos (individuais ou coletivos) ou indiretos aos participantes da pesquisa

O benefício de participação do aluno está na oportunidade de desenvolver o pensamento computacional, melhorando a resolução de problemas e capacidade de desenvolver seus próprios aplicativos, de acordo com sua necessidade laboral ou pessoal, ou seja, amplificando a conduta profissional mais eficiente e competente.

Previsão de riscos ou desconfortos

Os riscos possíveis de acontecer são:

- a invasão de privacidade/cansaço/aborrecimento: o aluno ao responder o questionário "Conhecendo Você", pode se sentir incômodo frente às questões. Para isso, o pesquisador, disponibiliza o questionário on-line, para que o sujeito escolha o momento mais conveniente para responder. E quanto a privacidade, as questões não são obrigatórias, ou seja, o sujeito responde conforme sua vontade. Além disso, também se oferece ao sujeito que não se sente confortável em responder as perguntas, um encontro com o pesquisador para dirimir dúvidas



sobre suas dúvidas será agendado. Caberá ao pesquisado a opção de participar, continuar ou desistir de participar.

- Divulgação de dados confidenciais. Os dados serão coletados em uma plataforma (Ambiente virtual de aprendizagem) que se encontra hospedada em um servidor instalado a partir de um IP (Protocolo da Internet) de uma instituição federal. A fim de minimizar este risco, será implementado diferentes níveis de acesso de usuários e backup dos arquivos, a nível de software.

Sobre divulgação dos resultados da pesquisa

Os resultados desta pesquisa serão divulgados em artigos científicos da área de computação e ensino. Também estes dados serão disponibilizados no AVA, a fim de facilitar o seu acesso.

Observações:

1. Este termo será assinado pelo aluno e pela pesquisadora, em duas vias de mesmo teor e forma;
2. A primeiras páginas deverão ser rubricadas.

"Em caso de dúvida quanto à condução ética do estudo, entre em contato com o Comitê de Ética em Pesquisa da Univates (Coep/Univates). O Comitê de Ética é a instância que tem por objetivo defender os interesses dos participantes da pesquisa em sua integridade e dignidade e para contribuir no desenvolvimento da pesquisa dentro de padrões éticos. Dessa forma o comitê tem o papel de avaliar e monitorar o andamento do projeto de modo que a pesquisa respeite os princípios éticos de proteção aos direitos humanos, da dignidade, da autonomia, da não maleficência, da confidencialidade e da privacidade. Contatos: (51) 3714.7000, ramal 5339 e coep@univates.br."



Walkiria Cordenonzi

walkiriacordenonzi@ifsul.edu.br – 55 55 999753329 – Paul Harris, 410. S. Livramento –RS –
Brasil

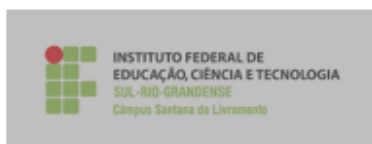
Santana do Livramento/Rivera, de de 2019.

Declaro que entendi os objetivos e condições de minha participação na pesquisa e concordo em participar.

(Assinatura do participante da pesquisa)

Nome do participante por extenso:

APÊNDICE D- Carta ao Diretor da CETP-UTU



Santana do Livramento, 20 de Agosto de 2018.

Señor Director de ETSR-UTU-Rivera
Profesor Ricardo Olivera

Por este meio solicito sua autorização para executar o Projeto Piloto denominado de "Eu programa 1.0!". Dito projeto faz parte das investigações da minha tese de doutorado na Universidade do Vale do Taquari – Brasil. O tema principal da tese é a observação do processo de ensino aprendizagem do Pensamento Computacional em adultos, que fizeram sua formação no Brasil ou no Uruguai (discentes dos cursos binacionais).

O projeto consiste em oficinas para alunos dos cursos binacionais e está planejado, inicialmente, para acontecer nas aulas do curso de Técnico em Logística, juntamente com o professor responsável Robert Gómez, no mês de setembro do corrente ano.

Aguardo sua autorização para dar início as oficinas.

Atenciosamente.

Prof. Walkiria Cordenonzi
IFSUL – Câmpus Santana do Livramento
walkiriacordenonzi@ifsul.edu.br

APÊNDICE E- Carta ao Diretor do CSL – IFSUL



Santana do Livramento, 15 de Abril de 2019.

Carta de Anuência

Sr. Diretor do IFSUL – Câmpus Santana do Livramento
Prof. Dr. Celso Silva Gonçalves

Por este meio solicito autorização institucional para executar uma pesquisa, cujo procedimento técnico é estudo de caso. Esta faz parte das investigações da minha tese de doutorado na Universidade do Vale do Taquari – Univates. O tema principal da tese é a observação do processo de ensino e aprendizagem do Pensamento Computacional em sujeitos adultos, atualmente matriculados em cursos binacionais. O projeto consiste em oficinas para estes discentes, que acontecerão nas dependências do Instituto, sendo o mesmo registrado na Pró-Reitoria de Pesquisa, Inovação e Pós-Graduação, identificado por PD00181118/156.

Ressalto que as identidades dos participantes serão mantidas em absoluto sigilo de acordo com a Resolução nº 466/12 do Conselho Nacional de Saúde (CNS/MS), que trata da pesquisa envolvendo Seres Humanos. Salientamos ainda que tais dados serão utilizados somente para a realização deste estudo ou serão mantidos permanentemente em um banco de dados de pesquisa, com acesso restrito, para utilização em pesquisas futuras.

Agradeço sua atenção e fico à disposição para quaisquer esclarecimentos adicionais que se fizerem necessários.

- Concordo com a solicitação
 Não concordo com a solicitação

Prof. Dr. Celso Silva Gonçalves
DIREÇÃO GERAL
IFSUL - Câmpus
Santana do Livramento

Prof. Walkiria Cordenonzi
IFSUL – Câmpus Santana do Livramento
walkiriacordenonzi@ifsul.edu.br

Apêndice F – Correção dos Testes do EC1

Id do Aluno/ habilidade	Pré teste		Teste 02		Pós teste				
	3	2 3	235	345	12	123	12345	12345	145
A02		x		x			x		x
A03							x	x	ne
A04							x	x	ne
A06		x		x				x	x
A07								x	x
A08			x				x	x	ne
A09			x	x				x	ne
A10							x	x	
		não respondeu o teste							
x		resposta errada							
ne		opção Não Entendi							

Apêndice H - Resultado dos Testes por Habilidade no EC2

ID aluno	TT	TQ	H1/12	H2/6	H3/11	H4/10	H5/11	Total	%
A01	4	16	8	4	9	5	7	11	68,8
A02	4	16	8	4	9	6	7	11	68,8
A03	4	16	11	5	10	8	8	14	87,5
A04	4	16	8	3	10	6	8	12	75,0
A05	4	16	7	5	8	4	5	10	62,5
A06	4	16	9	3	8	7	7	11	68,8
A07	2	9	7	0	7	6	6	8	50,0
A08	4	16	9	3	7	7	7	11	68,8
A09	4	16	7	1	5	4	6	5	31,3
A10	4	16	10	4	9	7	7	12	75,0
A11	3	12	4	3	4	2	2	5	31,3
A12	4	16	7	3	7	5	6	9	56,3
A13	4	16	7	3	8	4	4	9	56,3
A14	4	16	11	4	9	8	8	13	81,3
A15	4	16	7	4	7	5	7	10	62,5
A16	4	16	10	4	10	8	9	13	81,3
A17	3	11	4	2	6	3	4	7	43,8
A18	4	16	9	4	8	6	7	12	75,0
A19	3	14	6	1	7	5	5	8	50,0
A20	4	16	10	4	9	7	8	13	81,3
A21	2	7	1	1	1	0	0	1	6,3

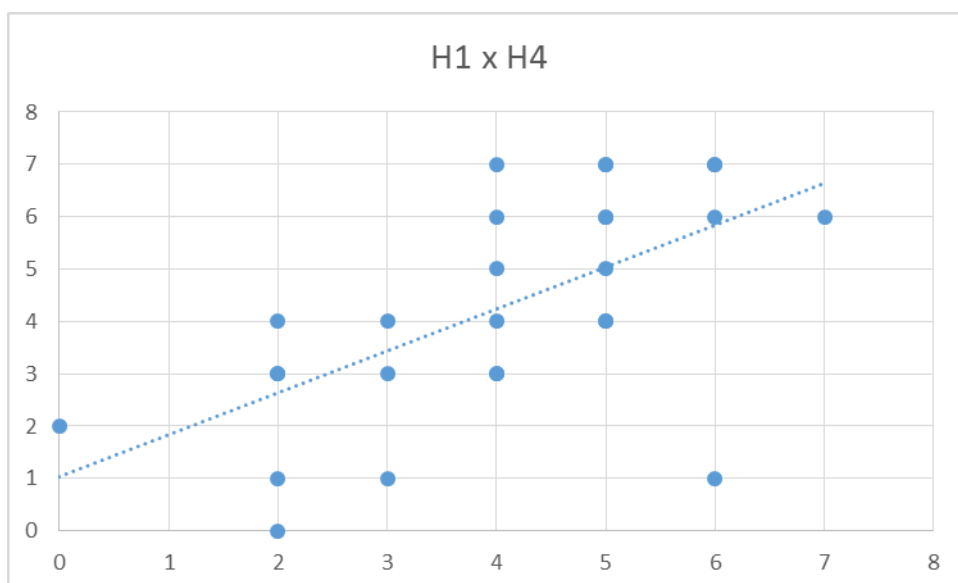
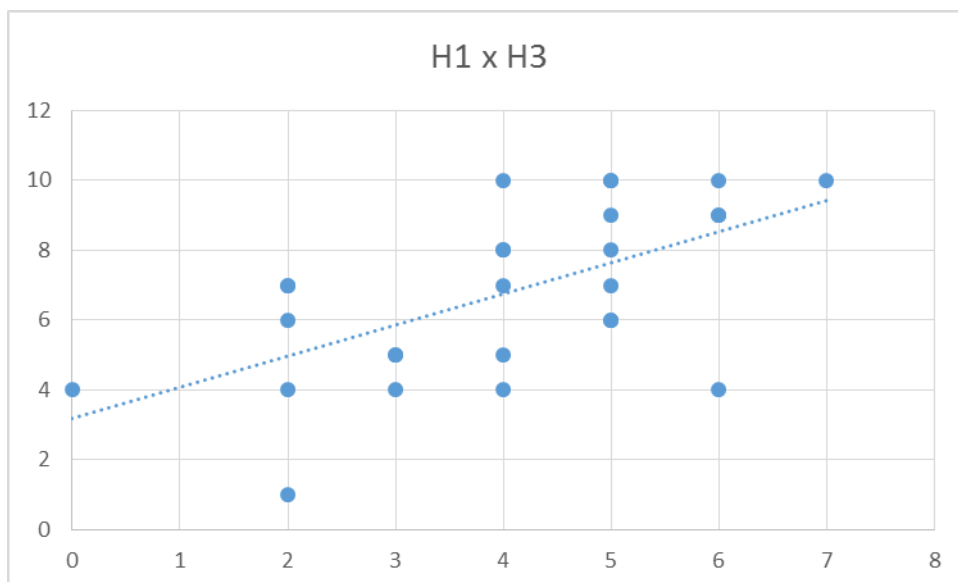
Fonte: autora (2020)

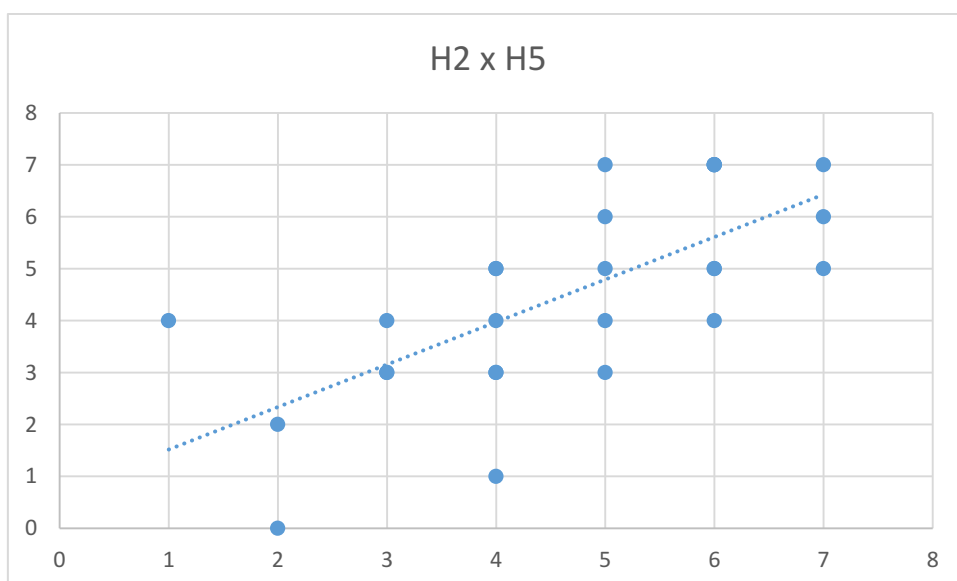
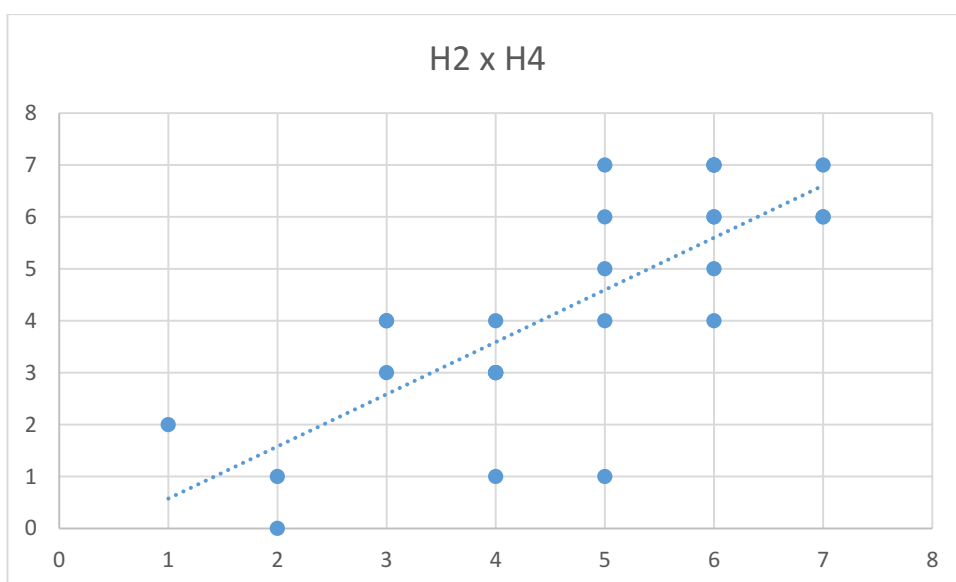
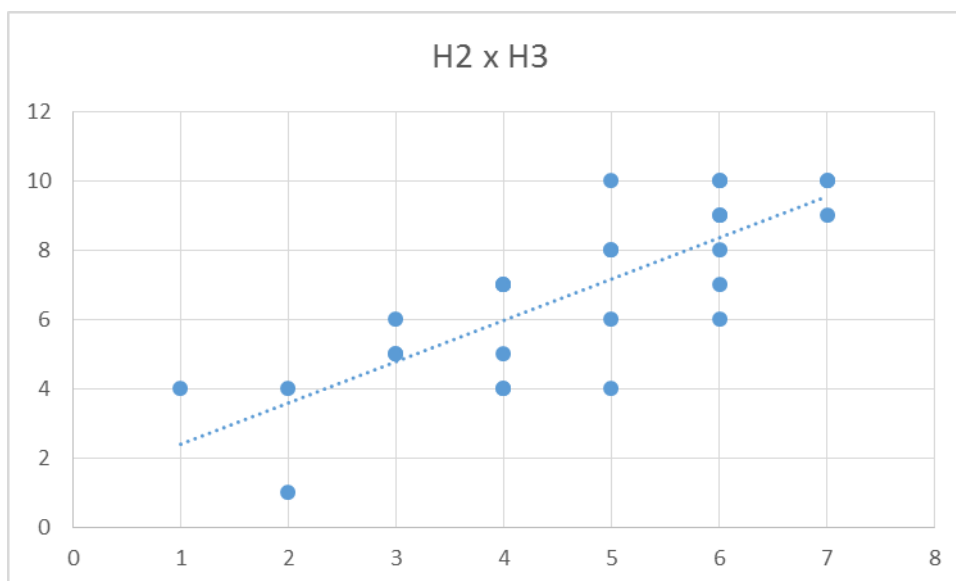
APÊNDICE J – Correção do Projeto Final: EC2

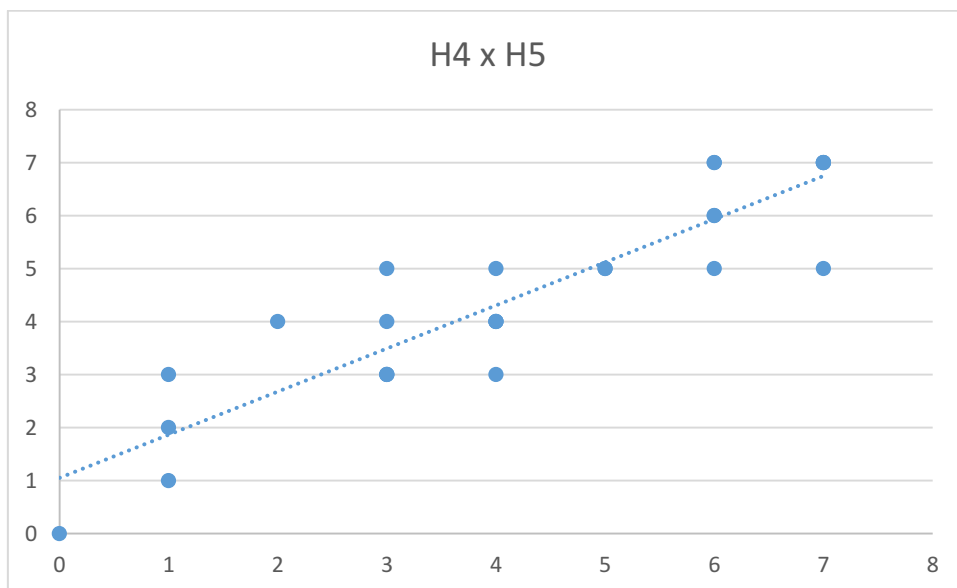
ID Aluno	Propõe-se a fazer o que é apropriado? Tudo o que foi planejado?	Faz o que foi proposto de forma correta?	Propõe-se a fazer o que é apropriado?	Atendeu suas necessidades?	É fácil de aprender?	É fácil de operar e controlar?	Tem validação de dados de entrada?	A interface é agradável?	É fácil de testar?
A01							NC		
A02									x
A03							NC		
A04	x	x		x			x	x	
A05							NC		
A06	x	x		x		x	NC		x
A07	x	x		x			NC	x	
A08							NC		
A09		x		x	x	x	NC	x	
A10		x					NC		
A11		x		x			x	x	
A12									
A13							NC		
A14							NC		
A15		x	x						x
A16							NC		
A17							x	x	x
A18	x	x		x	x		NC	x	x
A19			x				NC		x
A20		x							x
A21		x					NC		
NC	Não considerado								
x	resposta errada								

APÊNDICE K – Notas por Nacionalidade

	EC	Id Aluno	Nota
BR	1	A02	66,78
		A10	38,67
	2	A04	49,98
		A05	83,33
		A09	24,31
		A11	44,42
		A12	72,22
		A13	72,22
		A15	66,63
		A16	77,08
		A17	49,97
		A18	51,39
UY	1	A04	45,17
		A06	66,78
		A08	66,83
		A09	72,33
	2	A01	77,78
		A02	72,18
		A03	88,89
		A06	40,97
		A07	36,11
		A08	77,78
		A10	71,53
		A14	83,33
		A19	48,61
		A20	77,74
A21	49,31		

APÊNDICE L – Diagramas de Dispersão das Habilidades







UNIVATES

R. Avelino Talini, 171 | Bairro Universitário | Lajeado | RS | Brasil
CEP 95914.014 | Cx. Postal 155 | Fone: (51) 3714.7000
www.univates.br | 0800 7 07 08 09