

ENSINO DO PENSAMENTO COMPUTACIONAL NA EDUCAÇÃO BÁSICA

TEACHING OF COMPUTATIONAL THINKING IN BASIC EDUCATION

Lucas Eduardo Schlögl, Gabriel Castellani de Oliveira, Gian Carlo Giovanella, Artur Bizon, Bruno Santos, Nathan Kruger, Pedro Bursoni, Cedrik Buerger Neumann, Emiliane Eli Huber, Luciana Pereira de Araújo, Mauro Marcelo Mattos, Fabrícia Durieux Zucco, Karina Zendron, Nelson Hein

Laboratório de Desenvolvimento e Transferência de Tecnologias (LDTT)
Universidade Regional de Blumenau (FURB)
Blumenau, Santa Catarina, Brasil

{leschlogl, gabrielcastellani, gcgiovanella, abizon, bffsantos, nkruger, pbursoni, cbneumann, ehuber, lpa, mattos, fabricia, karinazedron, hein}@furb.br

Abstract. *Cognition is a process of knowledge acquisition based on sensory experiences, thoughts, representation and records [1]. The cognitive development is related to the improvement of the abilities to reasoning skills, decision-making, memorizing and realizing the environment, which in its turn are related to the fundamental practices of computer programming. Since the Sociedade Brasileira de Computação (SBC) argues that the teaching programming logic in elementary school is fundamental. We saw the possibility to stimulate this skill with game programming practices. This paper presents the developed platform, as well the experience acquired so far. In general, it can be noticed that the game increased the cognitive perception, the computational thinking, and the digital inclusion of the majority of the children as well.*

Resumo. *A cognição é um processo de aquisição do conhecimento que se baseia em experiências sensoriais, pensamentos, representações e recordações [1]. O desenvolvimento cognitivo está relacionado ao aprimoramento das habilidades de raciocinar, tomar decisões, memorizar e perceber o ambiente, que por sua vez se relacionam as práticas fundamentais da programação de computadores. Visto que a Sociedade Brasileira de Computação (SBC) defende que é fundamental o ensino da lógica de programação na educação básica, viu-se a possibilidade de estimular esse desenvolvimento com práticas de programação de jogos. Este trabalho apresenta a plataforma desenvolvida, assim como a experiência adquirida até o momento. De forma geral, pode-se notar que o jogo permitiu a evolução cognitiva da maioria das crianças, o pensamento computacional, assim como possibilitou a inclusão digital das mesmas.*

I. INTRODUÇÃO

De acordo com a Sociedade Brasileira de Computação (SBC) é fundamental o ensino da lógica de programação em escolas de nível fundamental [2]. Conforme SBC [2], “Não

é mais possível imaginar uma sociedade na qual os indivíduos não necessitem conhecimentos básicos de Computação, tão importantes para a vida na sociedade contemporânea quanto os conhecimentos básicos de Matemática, Filosofia, Física, ou outras ciências”. A computação permite que os indivíduos evoluam seus saberes relacionados ao desenvolvimento cognitivo, como as habilidades de raciocinar, tomar decisões estratégicas, memorizar e perceber o ambiente [1]. Essas habilidades são fundamentais para o viver do indivíduo, pois estão relacionadas as atividades cotidianas e que são enfrentadas ao longo da vida. Como exemplo tem-se a simples memorização do trajeto da escola-casa e vice-versa, assim como o raciocínio lógico utilizado para pagar alguém ou fornecer o troco. Ainda, a possibilidade da inclusão da lógica de programação em escolas básicas, aumenta a inclusão digital, uma vez que os laboratórios de informática são utilizados para esse ensino-aprendizagem.

Dentro desse contexto, tem-se como objetivo aprimorar e permitir o desenvolvimento desses saberes em um ambiente informatizado, possibilitando também a inclusão digital. Sabe-se que existem várias plataformas de ensino com esta finalidade, porém, a maioria delas é dependente de internet e não contempla todos os recursos para o ensino-aprendizado na língua Portuguesa. Visto que muitas escolas não possuem boa infraestrutura de rede, assim como a falta de funcionalidades no idioma Português, foi desenvolvido um jogo de computador voltado para o ensino da programação, no qual as crianças aprendem através do conhecimento construtivo. Isto é, em um primeiro momento elas brincam com o jogo, sendo que o mesmo apresenta o código-fonte gerado para a instrução executada. Em um segundo momento, as crianças digitam o código-fonte afim de visualizar os passos no jogo. Dessa forma, tem-se uma relação do pensar, raciocinar e memorizar.

Este trabalho é resultante de um projeto de extensão envolvendo os cursos de Ciência da Computação, Sistemas de Informação, Publicidade e Propaganda, Matemática e Letras. Dessa forma, para a construção de todo o ambiente informatizado foram unidos esses saberes em busca do aperfeiçoamento do leiaute, de um vocabulário adequado, assim como problemas matemáticos voltados para a faixa etária a ser trabalhada.

Nesse contexto, o artigo segue dividido na seguinte forma. A seção II apresenta os trabalhos relacionados que foram tomados como base para o desenvolvimento do jogo aqui apresentado, sendo o Code.org, o Greenfoot e o Scratch. A seção III aborda a respeito da inclusão da programação na educação básica como forma de ampliar o desenvolvimento cognitivo. A seção IV descreve o jogo desenvolvido, suas etapas, características e relaciona-o com os trabalhos relacionados. A seção V discorre sobre a experiência obtida até o momento através da aplicação do jogo em sala de aula com alunos no ensino fundamental de 3º e 4º ano de uma escola pública estadual. Por fim, a seção VI discute os resultados alcançados até o momento e relaciona os trabalhos futuros.

II. TRABALHOS RELACIONADOS

Essa seção apresenta três dos trabalhos tomados como base para o desenvolvimento do jogo apresentado neste artigo, sendo o Code.org, Greenfoot e Scratch. Cada um foi estudado de modo a orientar o desenvolvimento em um ponto específico.

O Code.org é uma plataforma lançada em 2013 dedicada para expandir o ensino da Ciência da Computação nas escolas, tendo em vista que a ciência da computação ajuda nas habilidades de resolução de problemas, na lógica e também na criatividade [3]. A plataforma trabalha inicialmente com o conceito de programação por blocos, tendo em

vista que o público inicial da plataforma tem uma faixa etária a partir de 4 anos, como é representado na Fig. 1. Para o público mais avançado da plataforma (como universitários), existe a possibilidade de utilizar linhas de comandos que são próximas a fala humana, como andar para frente (moveForward()), por exemplo, assim como representado na Fig. 2.

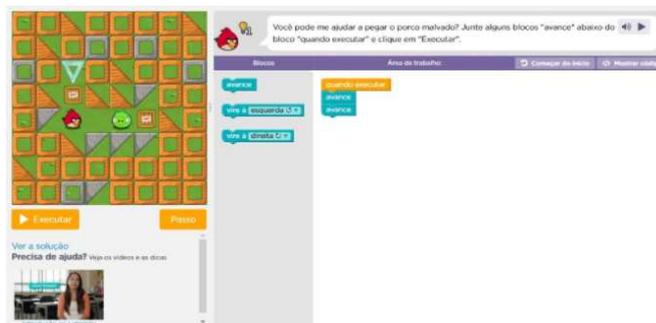


FIG. 1. CODE.ORG COM BLOCOS

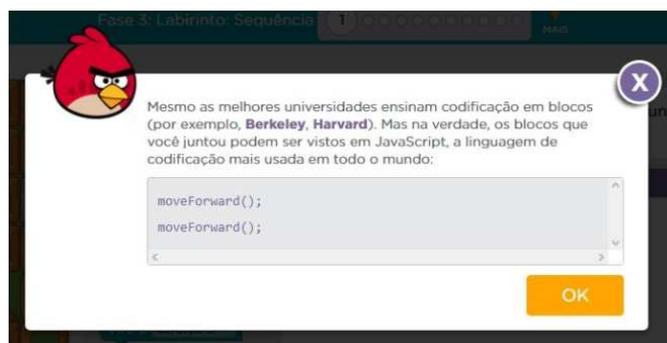


FIG. 2. CÓDIGO-FONTE LINGUAGEM NATURAL

Além do ensino da programação, o Code.org traz vários exercícios de lógica no formato múltipla-escolha para o aluno treinar a sua lógica de resolução dos problemas, como apresentado na Fig. 3. Tais exercícios podem ser impressos para que sejam resolvidos sem um computador.

Apesar das descrições e explicações dos conteúdos serem traduzidas, os comandos para a resolução dos exercícios ainda são em inglês.

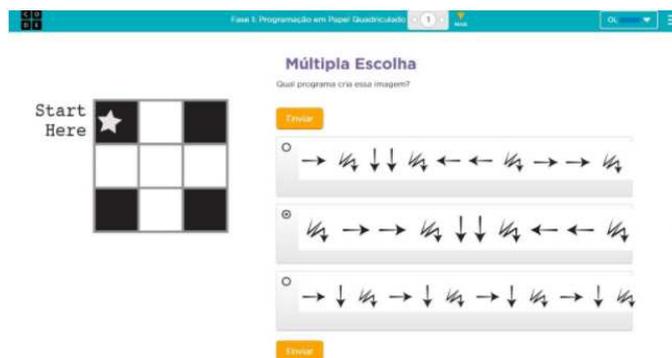


FIG. 3. EXERCÍCIOS DE LÓGICA

O segundo trabalho é o Greenfoot que é um programa para ensinar programação de forma fácil e divertida [4], tendo como foco ensinar a orientação a objeto na linguagem de programação Java. O programa é composto por um cenário denominado mundo que é dividido em células através de linhas e colunas, os quais recebem objetos denominados atores. No Greenfoot, o usuário interage com os atores e o mundo através de linhas de comando que são exibidos de maneira visual ao estudante, conforme apresentado na Fig. 4.

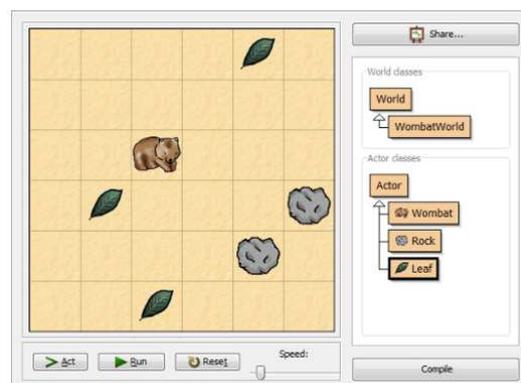


FIG. 4. CENÁRIO DO GREENFOOT

Apesar de tentar ser intuitivo aliando a programação com uma demonstração visual e com comandos próximos a fala humana, os comandos são em inglês, como por exemplo o comando descrito na Fig. 5 para virar para esquerda é traduzido como `turnLeft()`.

Por padrão, o ator principal do Greenfoot é o Wombat que é um marsupial da Austrália, porém o mesmo pode ser alterado pelo jogador. O Greenfoot foi construído para que crianças aprendessem a programar [4]. Devido a isso, sempre que um ator recebe um comando, o mesmo nunca é finalizado. Por exemplo, se for atribuído o comando `move()` para o Wombat, ele ficará andando do início ao fim do programa. Sendo assim, os comandos de repetição `for` e `while` são trabalhados somente em contextos nos quais há a repetição de um processo humano.

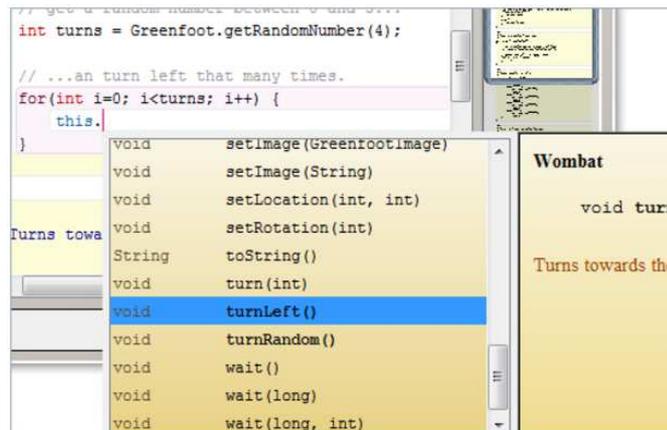


FIG. 5. COMANDOS DO GREENFOOT

O terceiro trabalho é o Scratch. Este se dispõe a ensinar a lógica de programação através da criação de histórias, jogos e animações [5]. A plataforma tem sua programação somente por blocos e seus comandos são traduzidos de uma maneira que se tenta assimilar os mesmos a fala humana.



FIG. 6. SCRATCH

Através da ferramenta, visualizada na Fig. 6, a criança pode criar seus próprios personagens e dar vida a eles. É possível trabalhar com movimentos, cores, sensores, condicionais, fluxos de repetição, entre outras características. Os blocos são organizados de forma a agrupar os comandos correspondentes a estas áreas. Após a programação dos blocos, o aluno pode iniciar o jogo e visualizar o resultado dentro da própria plataforma.

III. O ENSINO DA PROGRAMAÇÃO NA EDUCAÇÃO BÁSICA

Conforme o documento “Referenciais de Formação em Computação: Educação Básica” disponibilizado pela SBC [2] em julho de 2017 é fortemente recomendado que as escolas de educação básica tenham em sua estrutura curricular uma disciplina voltada ao ensino da programação. De acordo com ele, a computação é dividida em três eixos, como pode ser visualizado na Fig. 7, sendo o pensamento computacional, a cultura digital e o mundo digital.

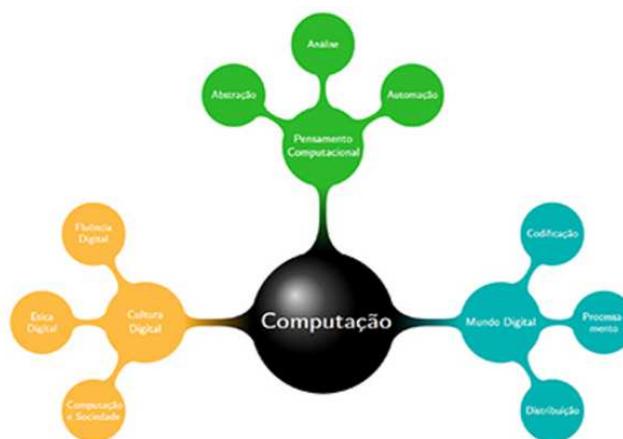


FIG. 7. EIXOS DA COMPUTAÇÃO

Cada um desses eixos é dividido em três segmentos, sendo: abstração, análise e automação para o pensamento computacional; fluência digital, ética digital e computação e sociedade para a cultura digital; e, codificação, processamento e distribuição para o mundo digital [2]. Conforme Garlet et al. [6], o pensamento computacional é o conjunto de habilidades voltadas a resolução de problemas adquiridos com o estudo de conceitos da Ciência da Computação. Outras fontes definem que o pensamento computacional é o processo cognitivo usado pelos seres humanos para resolver problemas [2] [7].

Entende-se como processo cognitivo as habilidades relacionadas as experiências sensoriais, pensamentos, representações e recordações [1] que são adquiridas durante o processo de desenvolvimento do ser humano. De acordo com Araújo et al. [7], a inclusão de disciplinas relacionadas ao ensino da lógica de programação nas séries iniciais, pode auxiliar no desenvolvimento do processo cognitivo das crianças.

Uma das possibilidades de incluir a lógica de programação nas séries iniciais é trabalhar com conceitos bases de forma lúdica, através de ferramentas e jogos [2] [7], como as descritas na seção II. Na literatura há alguns experimentos e projetos que aplicaram conceitos da programação através dessas ferramentas no ensino fundamental e obtiveram sucesso, como os trabalhos de Oliveira et al. [8].

As próximas seções descrevem o jogo produzido para o ensino da programação neste projeto, assim como o relato de experiência obtido com a prática na educação básica.

IV. DESENVOLVIMENTO DO JOGO PARA O APRENDIZADO DA PROGRAMAÇÃO

O desenvolvimento do jogo é dividido em duas etapas, sendo o jogo de tabuleiro e o jogo informatizado. Inicialmente, é apresentado o desenvolvimento do jogo de tabuleiro.

O jogo de tabuleiro foi desenvolvido com o objetivo de ser um suporte a explicação dos conteúdos para que o aluno utilize corretamente o jogo informatizado. O jogo de tabuleiro baseou-se no conceito apresentado pelo Code.org, sendo a ideia de apresentar conceitos antes de realizar a programação. Como atividade, baseou-se em uma atividade com cartas para realizar determinadas ações chegando a um objetivo, sendo desenvolvido um jogo de tabuleiro composto por um mapa e um conjunto de cartas com comandos como é apresentado na Fig. 8. No jogo de tabuleiro, adaptou-se o conteúdo e

os enunciados para que os alunos resolvessem os exercícios de lógica e programação utilizando como base questões que envolviam temáticas locais, como o mapa do estado de Santa Catarina e da cidade de Blumenau.

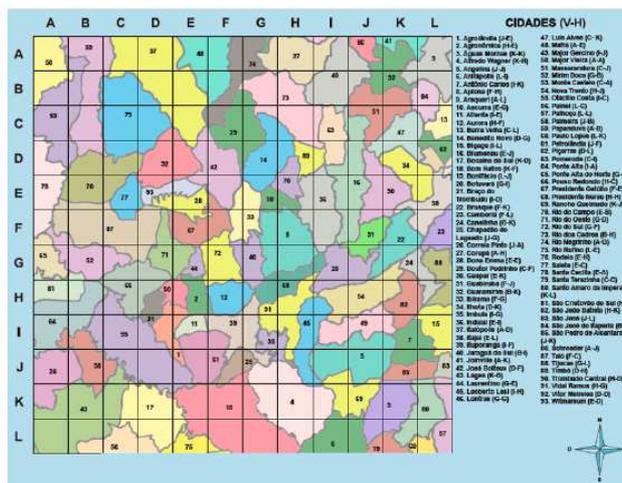


FIG. 8. JOGO DE TABULEIRO DESENVOLVIDO

Para resolver os exercícios, utilizam-se cartas com comandos de andar em uma direção, como por exemplo andarNorte(). Com essas cartas, os alunos precisam resolver o problema chegando em um local que o enunciado pede, como por exemplo no enunciado seguinte: “Você está em Lages e tem o desejo de ir até Blumenau, pois ouviu falar sobre a qualidade da Oktoberfest que acontece por lá. Porém, houve chuva intensa pelo estado nos últimos dias, fazendo com que você tenha que repensar seu trajeto até chegar no seu destino. Decidiu-se então que iria ser evitado passar pelas cidades de Ilhota, Indaial e Massaranduba. Defina qual será a melhor rota para que você chegue até Blumenau para aproveitar a festa.”

Assim, a cada carta que se utiliza para andar, o aluno precisa anotar qual carta utilizou, para que no final, chegue a um algoritmo para a resolução daquele problema. O jogo foi construído envolvendo toda a equipe interdisciplinar, de modo que o grupo de Publicidade e Propaganda cuidou do leiaute do tabuleiro e das cartas. O grupo de Computação aplicou a lógica de programação e as regras do jogo. O grupo de Letras criou os enunciados. E o grupo de Matemática auxiliou a composição dos enunciados junto com os comandos lógicos projetados.

Tendo em vista que o jogo de tabuleiro serviria somente para a introdução da programação e para a percepção dos conceitos cognitivos, em paralelo ao uso do jogo (apresentado na próxima seção), foi desenvolvido o jogo informatizado. O desenvolvimento do jogo informatizado é dividido em duas etapas, sendo a versão 1, a

qual os alunos interagem com o jogo pelas setas do teclado e a versão 2, a qual os alunos interagem com o jogo através de comandos de programação. Apesar de estar dividido em duas versões, as mesmas se integram nos níveis de dificuldade, sendo que a versão 1 é correspondente ao nível fácil e a versão 2 aos demais níveis.

O jogo foi desenvolvido sobre uma plataforma de ensino da programação já desenvolvida por professores do curso de Ciência da Computação da Universidade, que é utilizada para o ensino da Introdução a Programação com alunos da graduação. A escolha por utilizar a mesma plataforma foi para ganhar tempo com questões de *back-end*, além de ser utilizada há quase 10 anos nos cursos e obter um *feedback* positivo no ensino-aprendizagem da programação. Essa plataforma é semelhante ao Greenfoot, apresentado na seção II, sendo composta por um mundo dividido por células formados por linhas e colunas e objetos que se alocam a essas células. Os objetos disponíveis para este mundo são: robô (objeto principal), paredes (obstáculos), números, booleanos, alienígenas e tesouros. Esses últimos podem ser utilizados como obstáculos ou para a resolução de atividades matemáticas.

As versões atuais foram desenvolvidas em Java, por ser a linguagem de domínio da equipe de programação envolvida, bem como ser a linguagem da plataforma atual. Dentro do desenvolvimento foi necessário dividir o desenvolvimento em três partes essenciais, sendo: a renderização dos componentes gráficos; interação de objetos com o mundo; e por último, a configuração dos exercícios. A renderização dos componentes gráficos é responsável por mostrar graficamente o mundo do jogo e todas as ações que estão ocorrendo, como por exemplo, mostrar a movimentação do robô. A interação dos objetos no mundo cuida da parte de gerenciamento dos objetos, como por exemplo, onde cada objeto está, se esses objetos estão colidindo com algo, bem como as ações de andar de algum objeto. Por último, a configuração dos exercícios é responsável pela configuração das fases, enunciado e o mundo, assim como pelo acompanhamento pedagógico, tendo em vista que a solução proposta pelo aluno para um exercício é persistida através de e-mail, para que seja possível verificar como os mesmos estão resolvendo os exercícios.

Para o desenvolvimento das duas versões, novamente contou-se com toda a equipe interdisciplinar da seguinte forma. A equipe de Publicidade ficou responsável por desenhar a interface do jogo, assim como redesenhar os objetos a serem colocados no mundo. Para o desenho baseou-se em jogos de computadores antigos que tem como foco a atividade a ser completada. A equipe de matemática ficou responsável por criar os enunciados relacionados aos cálculos. A equipe de Letras ficou responsável por validar os enunciados, assim como fazer um estudo a respeito do letramento digital dessas crianças. Por fim, a equipe de programação ficou responsável por programar o contexto projetado.

Após o desenvolvimento, obteve-se uma versão que ao iniciar o jogo, o aluno se depara com a seleção de dificuldade dos exercícios, conforme demonstrado na Fig. 9. Isso é necessário pelo fato de que quanto mais difícil o exercício for, mais conhecimento sobre lógica e programação o mesmo precisará ter. Isso se assemelha ao que é proposto pelo Code.org, sendo que o curso é separado por aulas e em cada aula o aluno resolve os exercícios que são recomendados para treinar aquele conhecimento e quanto mais aulas o aluno realiza, maior é a dificuldade do exercício proposto.



FIG. 9. FASES DO JOGO

Ao escolher uma fase, o aluno se depara com vários exercícios enumerados (Fig. 10) e ao escolher um deles é mostrado o enunciado do exercício (Fig. 11), isto é, o que o mesmo deve fazer para completar aquele exercício. Ao entender o que se pede, o aluno pode prosseguir para a resolução do exercício. Um pensamento semelhante a este é adotado pelo Code.org, sendo que após escolher o exercício, é apresentada a tela do jogo com o enunciado sobre o local no qual é inserida a lógica para resolução do exercício.



FIG. 10. EXERCÍCIOS DENTRO DE UMA FASE

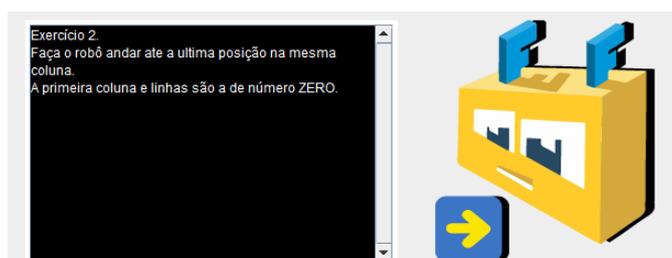


FIG. 11. ENUNCIADO DE UMA ATIVIDADE

A parte principal do jogo, que é a resolução dos exercícios, traz uma interface limpa a qual exhibe somente o essencial, como o mundo, o robô, os objetos de interação para aquela atividade e alguns comandos, conforme disposto na Fig. 12.

Inicialmente, na versão 1, o jogador executa a ação, como por exemplo, fazer com que o robô chegue ao tesouro. Para isso ele deve utilizar as setas do teclado e quando chegar ao destino, deve utilizar o botão “diga” para apresentar que completou a missão. Após finalizar, o aluno tem acesso ao código-fonte que foi gerado pelo robô, como apresentado na Fig. 13.



FIG. 12. CENÁRIO DO JOGO

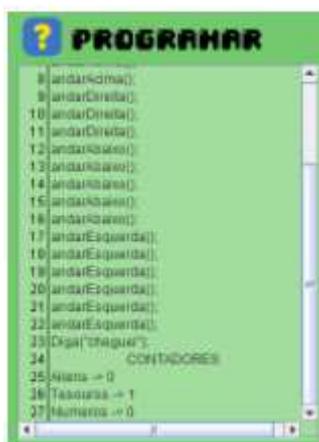


FIG. 13. CÓDIGO-FONTE GERADO

O código gerado, assemelha-se ao que é proposto pela plataforma Code.org, sendo que o aluno realiza os comandos através de blocos e tem um código gerado semelhante a fala humana. Também é similar ao Greenfoot, no qual só é possível resolver os exercícios através de comandos que tendem a simular a fala humana. O que difere do jogo proposto a esses trabalhos, é que os códigos gerados são totalmente em português, facilitando o entendimento pelos alunos.

Após o código-fonte gerado, também se tem os valores dos contadores que são utilizadas como suporte na conclusão do objetivo do exercício, introduzindo conceitos de variáveis. Esses contadores também são atualizados em tempo de execução sempre que o robô passa por cima de um alienígena, tesouro ou número.

Após a conclusão da versão 1 e de seu uso foi desenvolvida a versão 2 do jogo. Em meio a isso, foi construído um segundo jogo de tabuleiro, o qual possui um fluxograma com diversas condicionais, e o aluno deve seguir o fluxo do início ao fim. O objetivo desse fluxo foi recapitular conceitos aprendidos com a versão 1 e introduzir o conceito de tomada de decisão.

Dessa forma, a versão 2 do jogo, além de aceitar a interação com o robô diretamente por comandos em linhas de código-fonte, como andarDireita(),

andarEsquerda(), andarAcima() e andarAbaixo(), sofreu alterações de interface gráfica e funcionalidades. As funcionalidades novas programadas foram relacionadas a cálculos matemáticos e a comandos de repetição.

A tela de cálculo tem como objetivo auxiliar em contas matemáticas envolvendo vários números, fazendo com que o aluno se preocupe mais com a lógica para resolver o exercício. Para utilizar essa funcionalidade, o aluno deve estar em um cenário que contenha vários números. Dessa forma, ele terá que passar sobre os números capturando-os. Cada número capturado é adicionado na tela de cálculo ficando um espaço em branco entre eles. Sendo assim, o aluno poderá inserir as operações necessárias para a resolução do cálculo matemático. Essas operações serão dispostas entre os números e ao final é obtido o resultado, conforme pode-se observar na Fig. 14.

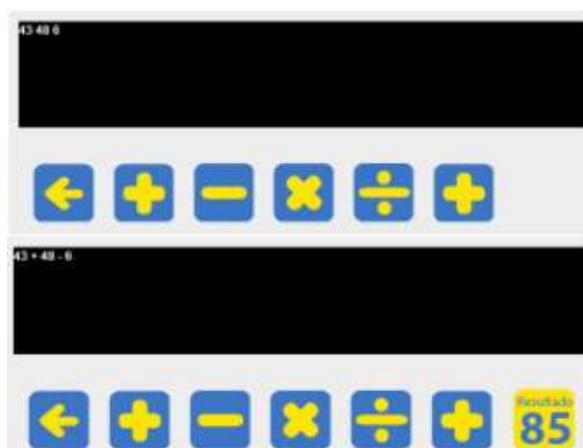


FIG. 14. TELA DE CÁLCULO

Já a tela repetição tem como auxílio a utilização de estruturas de repetição fornecendo uma abordagem semelhante ao que o Scratch utiliza para suas estruturas de repetição, ou seja, através de blocos. A ideia é que o aluno selecione uma condição dentre as disponíveis e dentro do quadro escreva o código-fonte sequencial, assim como demonstrado na Fig. 15. Após a programação do código-fonte nas telas de programar, independente de possuir laço de repetição ou não, o aluno pode executar o código-fonte e visualizar o robô executando sua missão. Caso o aluno escreva algum código de forma incorreta, a ferramenta o informa. Desse modo são trabalhados os conceitos cognitivos de memorização e raciocínio lógico.

Com essas características, realizou-se um quadro comparativo entre o jogo proposto na versão 2 e os trabalhos correlatos, que pode ser visto no QUADRO 1. Através da figura pode-se observar que as características uso de tabuleiro para o ensino dos conteúdos, código semelhante a fala humana, estrutura de blocos para programação, ferramenta em português, código gerado em português, estrutura de repetição em forma de blocos e auxílio a cálculos foram atendidas pelo jogo desenvolvido. Por outro lado, as ferramentas disponíveis no mercado, além de serem oferecidas de forma gratuita permitem que o educador crie seus próprios exercícios, fornecendo assim uma maior flexibilidade, sendo esta uma característica a ser desenvolvida.



FIG. 15. TELA DE PROGRAMAÇÃO

QUADRO 1. COMPARAÇÃO ENTRE OS CORRELATOS E O JOGO DESENVOLVIDO

	CODE.ORG	GREENFOOT	SCRATCH	JOGO PROPOSTO
Utilização de tabuleiro para o ensino de conteúdos	X			X
Código semelhante a fala humana	X	X		X
Estrutura de blocos para programação	X		X	
Ferramenta em português (blocos e enunciados)	X		X	X
Código gerado em português				X
Estrutura de repetição em forma de blocos	X		X	X
Auxílio a cálculos				X

V. RELATO DE EXPERIMENTO EM CAMPO

Por se tratar de um projeto de extensão e para validar os conceitos estudados do jogo desenvolvido foi escolhida uma escola da rede pública que manifestou interesse em receber a prática e que disponibilizou horário de aula e espaço em laboratório para que a atividade fosse realizada. Dessa forma, a prática iniciou em fevereiro de 2017 e permanece até a data da elaboração deste artigo. O projeto é executado com alunos do

3º e 4º ano do ensino fundamental, sendo turmas dos períodos matutino e vespertino respectivamente, com carga horária semanal de uma hora e turmas compostas por uma média de 20 alunos cada, sendo um total de 42 alunos.

O experimento foi realizado da seguinte forma: em um primeiro momento foi introduzido o conceito de “passos sequenciais” através do primeiro jogo de tabuleiro desenvolvido no projeto; em seguida foi trabalho com a versão 1 do jogo informatizado; após isso, voltou-se para outro jogo de tabuleiro com objetivo de ensinar o conceito de “tomada de decisão” e “variáveis”; por fim, no último momento trabalhou-se com a versão 2 do jogo informatizado. Cada uma dessas etapas é relatada a seguir.

A. Etapa 01: jogo de tabuleiro 01

A fim de trabalhar os saberes cognitivos voltados ao raciocínio lógico, sequencial e a memorização foi iniciada a prática com o jogo de tabuleiro desenvolvido para o aprendizado da lógica sequencial. Como apresentado na seção anterior, o jogo possui as características já trabalhadas pelos alunos, como o mapa da cidade/estado e a rosa dos ventos. Dessa forma, os alunos já estão situados no ambiente, ficando para o aprendizado somente a parte computacional.

Nessa etapa, as crianças foram divididas em equipes de quatro alunos cada, conforme apresentado na Fig. 16. Cada equipe deveria individualmente resolver algumas atividades que eram proporcionadas, sendo que o objetivo sempre era o mesmo: sair de um bairro ou cidade de origem e chegar em um bairro ou cidade de destino. Para que as crianças alcançassem este objetivo, elas precisavam percorrer células do tabuleiro, dispostas em linhas e colunas. Para percorrê-las, elas recebiam as cartas contendo as direções (norte, sul, leste, oeste) e tinham que as disponibilizar em uma sequência lógica, que permitisse a saída da cidade/bairro origem até o destino. Após a disponibilização das cartas na sequência, um professor conferia se estava correto e permitia então, que passassem para a próxima etapa.



FIG. 16. APLICAÇÃO DA ETAPA 1

Essa etapa durou cerca de 4 semanas e permitiu que as crianças desenvolvessem suas habilidades em resolver problemas, ligadas diretamente ao pensamento computacional. Para cada tarefa, elas tinham que analisar o caminho, abstrair os dados relacionados a direção e então projetá-los através de um fluxo sequencial. Esse fluxo sequencial, por sua vez, era o resultado da automatização do processo, pois sempre apresentava o caminho entre a origem e o destino.

Ainda, embora as crianças não tenham esse conhecimento, o processo sequencial produzido no final de cada atividade, correspondia a um algoritmo. Esse algoritmo produzido de forma lúdica, é uma das atividades definidas pelo documento de “Referenciais de Formação em Computação: Educação Básica” [2] que deve ser aplicado no ensino fundamental. Sendo que a prática mostrou que realmente é possível sua aplicação com crianças desta faixa etária.

Com relação as crianças, de forma geral se mostraram bem animadas com as atividades, mas não viam a hora de programarem no computador. As crianças se sentiram confortáveis com a rosa dos ventos utilizada para indicar a direção, pois já conheciam ela da disciplina de geografia, o que facilitou que entendessem a sequência dos passos. Essas características podem ser afirmadas através da análise do discurso das crianças durante a execução da atividade. Essa análise do discurso foi feita pelos pesquisadores (bolsistas e professores) que anotavam as falas das crianças durante as atividades realizadas. Posteriormente os pesquisadores se reuniram para a avaliação e análise dos dados coletados, chegando a esta conclusão como relatado.

B. Etapa 02: versão 01 do jogo informatizado

Após trabalhar com o jogo de tabuleiro, as crianças foram levadas para um laboratório no qual podiam trabalhar em duplas ou individualmente, como apresentado na Fig. 17. Essa etapa durou cerca de dois meses e foi segmentada em vários objetivos de aprendizagem. Como objetivo inicial foi trabalhar com o algoritmo seguindo um fluxo sequencial por meio de setas do teclado. Dessa forma, foi disponibilizado o jogo desenvolvido e o mesmo foi explicado. Como era algo novo para as crianças, em um primeiro momento elas prestaram atenção e então executaram os passos conforme indicado. Para dar a impressão de continuidade da etapa 1 para a 2, trabalhou-se com o cenário composto por linhas e colunas e com os comandos `andarNorte()`, `andarSul()`, `andarLeste()` e `andarOeste()` assim como nas cartas. Em seguida os comandos foram alterados para direções: esquerda, direita, abaixo e acima.

Algumas crianças tiveram dificuldades por não estarem familiarizadas com o computador, porém todas já tiveram contato alguma vez com a máquina. Como o programa é dividido em três níveis de dificuldade (fácil, médio e difícil), foi instruído que nas três primeiras aulas as crianças trabalhassem somente com o nível fácil. Nesse nível haviam dois tipos de atividades, sendo labirintos, no qual o robô saía de um local da tela e precisava chegar até o destino representado por um tesouro e indicação de caminhos que tinham que ser percorridos no cenário, como andar ao redor do mundo, andar em ziguezague (passar pela primeira linha da esquerda para a direita, pular para a linha de baixo e voltar da direita para a esquerda continuando esse fluxo até chegar ao final do tabuleiro), entre outros. Para a maioria, os cenários relacionados aos labirintos foram fáceis de percorrer. Já os relacionados a um caminho para percorrer eram um pouco mais difíceis.

Após algumas atividades, verificou-se que as crianças tinham dificuldades nos caminhos sem labirinto, pois não liam o enunciado, logo, não sabiam o que fazer. Ainda, alguns enunciados, quando lidos, apresentavam termos difíceis para a compreensão das mesmas. Para tentar solucionar o problema, o enunciado foi colocado em uma tela antes da inicialização do jogo, assim como revisados.



FIG. 17. APLICAÇÃO DA ETAPA 2

Dessa forma, ao trabalhar o jogo novamente com as crianças elas liam e então executavam o que se pedia. Algumas continuavam não lendo o enunciado, pois achavam que já sabiam o que fazer, sendo este um problema cultural que não foi solucionado pelo projeto.

Para um segundo objetivo dessa etapa, foram construídos cenários que envolviam outros objetos como alienígenas, tesouros e números. Esses cenários foram disponibilizados na fase intermediária do jogo. O objetivo desses cenários era desenvolver as habilidades de tomar decisões e raciocinar antes de agir. Um exemplo de atividade era a seguinte: cenário composto por vários números, robô inicia em uma posição aleatória do mundo e precisa passar uma única vez por todos os números pares do cenário, e caso não conseguir, deve informar que não alcançou o objetivo. Para a resolução dessa atividade, a criança precisava pensar antes de agir, ou seja, precisava saber quais são os números pares e qual o melhor caminho que poderia fazer para não passar mais de uma vez sobre o mesmo número, assim como para não passar sobre um número ímpar.

Na prática, a resolução dessa atividade foi mais difícil e trabalhosa para alguns, enquanto outros conseguiram resolvê-las rapidamente igualmente as fases iniciais. Esses que tiveram dificuldades, não entendiam que só podiam passar uma única vez pelos números e que estes tinham que ser par. Enquanto os alunos que compreenderam a atividade passaram para os próximos níveis, foi indicado aqueles que ainda não captaram bem que fizessem o exercício novamente. Dessa forma, cada criança pode aprender no seu tempo e evoluir o seu saber. Para as crianças com mais dificuldades haviam professores apoiando a atividade. Com o decorrer dos exercícios e a prática, observou-se que mesmo as crianças com mais dificuldade inicialmente conseguiram captar a ideia do jogo e conseguiram atingir os objetivos propostos, se animando na resolução dos demais.

Em toda a etapa foi indicado aos alunos que primeiro escrevessem os passos sequenciais no papel para depois utilizar as setas do teclado afim de executar o processo que realizaram no papel. Dessa forma, foi instigado o pensar antes de fazer. Para toda a atividade realizada, o jogo gerava o código-fonte resultante após a conclusão da atividade. Para aquelas crianças com maiores dificuldades, foi instruído que copiassem o

código gerado, para que pudessem compreender e analisar melhor o motivo de sua sequencial produzida.

Ainda nessa etapa da atividade, foram propostos alguns exercícios que iniciavam o conceito de variáveis, sendo um exemplo: existem números espalhados pelo cenário, o robô inicia na primeira célula do tabuleiro e sempre que chegar ao final da linha deve apresentar a soma dos números da linha a qual passou. Essa atividade foi bem difícil para quase todos os alunos, pois não entendiam que precisavam manter a soma em memória até chegar ao final da linha e quando passavam para a próxima linha tinham que começar a somar novamente. Dessa forma, decidiu-se construir um jogo de tabuleiro que ensine os conceitos de variáveis e junto a ele incluir a tomada de decisão de fluxo. Ainda, devido a este motivo foi construída a tela de cálculo da versão 2 descrita na seção anterior.

C. Etapa 03: jogo de tabuleiro 02

A etapa 03 iniciou no retorno das férias de julho dos alunos, possuindo um intervalo de duas semanas com relação a última etapa desenvolvida. Dessa forma, a atividade teve como objetivo inicial rever os conteúdos aprendidos até o momento e introduzir os conceitos de variáveis e tomada de decisão em um jogo de tabuleiro. Assim como o primeiro jogo de tabuleiro, este teve por objetivo ensinar a parte conceitual, para que então os alunos voltassem ao jogo informatizado para aplica-los.

Assim como descrito na seção anterior, o jogo aplicado nesta etapa era composto por um fluxograma no qual os alunos tinham que encontrar o resultado de duas variáveis seguindo os passos do fluxo. Essa etapa durou 3 semanas, sendo que na primeira semana foram divididos em grupos de quatro alunos, na segunda semana trabalharam individualmente e na terceira semana trabalharam em duplas. Na Fig. 18 pode-se observar as crianças realizando a atividade.



FIG. 18. APLICAÇÃO DA ETAPA 3

As primeiras atividades foram difíceis de serem completadas, pois os alunos tiveram dificuldades de compreender que se optassem por um fluxo não precisavam voltar para o fluxo anterior, ou seja, aquele que ficou para trás. Ainda, alguns não compreenderam como trabalhar com a variável, porém após a realização de algumas

atividades, os alunos conseguiram compreender a ideia da atividade e conseguiram resolver de forma correta.

Percebeu-se que com o retorno das férias, alguns alunos haviam esquecido conceitos básicos, como o significado de par/ímpar, sendo que antes das férias eles possuíam esse conhecimento. Dessa forma, além dos conceitos do pensamento computacional foi necessário tratar conceitos matemáticos que haviam sido esquecidos.

D. Etapa 04: versão 02 do jogo informatizado

Após as três semanas trabalhando com o jogo de tabuleiro, voltou-se ao laboratório para a prática com o jogo informatizado. Dessa vez, as crianças deveriam programar o seu próprio código-fonte, escrevendo os passos sequenciais para que o robô andasse no tabuleiro e cumprisse sua missão. Para essa etapa foram disponibilizados dois cenários, sendo um tabuleiro em branco e outro tabuleiro contendo anjos (um novo objeto da versão 2), números e tesouros. Nessa versão foi introduzido o objeto anjo, sendo o amigo do robô e o objeto alienígena (já existente), sendo o inimigo do robô. No tabuleiro em branco as crianças tinham que percorrer ao redor do mundo e, em seguida, em um sentido diagonal. Já no segundo tabuleiro, as crianças tinham que passar por cima de todos os números e em seguida por cima de todos os anjos, uma única vez.

Atualmente o projeto se encontra nessa etapa e teve duração até o momento de três semanas. De forma geral, essa etapa foi surpreendente para os professores, pois as crianças conseguiram compreender o conceito da programação sequencial conseguindo digitar seu código-fonte na sequência correta para cumprir com o objetivo proposto pelo jogo.

VI. RESULTADOS E TRABALHOS FUTUROS

Este artigo apresentou o desenvolvimento de um jogo voltado para o ensino da programação, assim como a experiência alcançada após seis meses de uso e experimentação do mesmo. Do ponto de vista dos programadores e professores de computação envolvidos no processo, pode-se afirmar que os alunos conseguiram adquirir em pouco tempo, os conceitos bases da programação. Isso pode ser afirmado através do relato de experiência em campo, pois os alunos conseguem criar pequenos códigos-fontes que permitem os objetos do jogo se moverem de acordo com o objetivo do exercício. Ainda, durante a experiência relatada, foi possível perceber que alguns alunos conseguiram pensar além do disposto até o momento, como por exemplo, que para fazer com que o objeto se mova várias vezes para um mesmo sentido é necessária a repetição de um trecho de código-fonte. Essa questão na programação indica laços de repetição ou sub-rotinas, e ainda não foram abordadas com os alunos.

Do ponto de vista das pessoas envolvidas com o aspecto visual e teórico do jogo, pode-se afirmar que os alunos gostaram das mudanças visuais projetadas, assim como gostam de leiautes mais modernos e relacionados com o seu cotidiano. Como por exemplo, ao adicionar o objeto anjo no jogo, ficou mais evidente para os alunos que este é o amigo do jogador e que o objeto alienígena é o inimigo.

Do ponto de vista pedagógico relatado pelas professoras de educação básica desses alunos em uma conversa informal, pode-se afirmar que houve uma melhora na atenção e no pensar para raciocinar cálculos matemáticos, por exemplo. Ainda, conforme elas, o jogo contribui bastante para o desenvolvimento pessoal, pois ensina a tomada de decisões que serão necessárias no dia a dia das mesmas.

Além disso, pode-se notar uma evolução melhor entre o jogo de tabuleiro e a programação do que somente na programação. Houve uma tentativa de evoluir para a próxima etapa sem a inclusão do jogo de tabuleiro, porém, dessa forma as crianças tiveram mais dificuldade e não conseguiram resolver as atividades. Por isso, foi desenvolvido o segundo jogo de tabuleiro que trabalhava com variáveis e fluxos condicionais. No projeto, o jogo de tabuleiro teve objetivo de fornecer o aprendizado conceitual para que então no jogo informatizado, os conceitos fossem aplicados. Por outro lado, nada impediria que os conceitos fossem abordados também em um tabuleiro informatizado, podendo ocorrer a omissão do jogo de tabuleiro.

Como trabalhos futuros, será trabalhado com as questões relacionadas aos laços de repetição e variáveis. Também pretende-se criar uma interface para que os próprios alunos construam seus tabuleiros e enunciados, sendo que hoje os exercícios são entregues prontos dentro da própria plataforma do jogo. Ainda, tem-se a intenção de criar um acompanhamento da turma (semelhante ao Code.Org), de modo que o aluno abra somente os exercícios apropriados para o seu conhecimento.

ACKNOWLEDGMENT

Agradecemos a Universidade Regional de Blumenau (FURB) pelo apoio e financiamento do projeto desenvolvido. Também agradecemos a Escola de Educação Básica Pedro II por ceder o espaço e suas turmas para que o projeto fosse aplicado.

REFERÊNCIAS

- [1] W. WEISZFLOG. “Michaelis Moderno Dicionário da Língua Portuguesa”. São Paulo: Melhoramentos, 2004.
- [2] A. L. A. Raabe et al., “Referenciais de Formação em Computação: Educação Básica”, Workshop sobre Educação em Computação, Sociedade Brasileira de Computação (SBC), Julho, 2017. Disponível em: <<http://www.sbc.org.br/files/ComputacaoEducacaoBasica-versaofinal-julho2017.pdf>>. Acesso em: set. 2017.
- [3] Code.org. Aprenda no Code Studio. 2017. Disponível em: <<https://studio.code.org/courses>>. Acesso em: set. 2017.
- [4] University of Kent, Greenfoot, 2015. Disponível em: <www.greenfoot.org>. Acesso em: set. 2017.
- [5] Lifelong Kindergarten, MIT Media Lab, Crie estórias, jogos e animações e partilhe com gente de todo o mundo, 2017. Disponível em: <<https://scratch.mit.edu/>>. Acesso em: set. 2017.
- [6] D. Garlet, N. M. Bigolin, e S. R. Silveira, “Uma Proposta para o Ensino de Programação de Computadores na Educação Básica”, Departamento de Tecnologia da Informação, Universidade Federal de Santa Maria, RS, 2016.

[7] D. C. ARAÚJO, A. N. RODRIGUES, A. N.; C. V. SILVA, L. S. SOARES, “Ensino da Computação na Educação Básica Apoiado por Problemas: Práticas de Licenciados em Computação,” Anais do XXIII WEI (Workshop sobre Educação em Computação) Garanhuns, 2015. Disponível em:

<<http://www.lbd.dcc.ufmg.br/colecoes/wei/2015/014.pdf>> Acesso em: set. 2017.

[8] M. S. L. de OLIVEIRA, A. A. de SOUZA, A. F. BARBOSA, E. F. S. BARREIROS, “Ensino de lógica de programação no ensino fundamental utilizando o Scratch: um relato de experiência,” XXXIV Congresso da Sociedade Brasileira de Computação (CSBC), 2014. Disponível em:

<<http://www.lbd.dcc.ufmg.br/colecoes/wei/2014/0022.pdf>>. Acesso em: set. 2017.