



Um Ambiente Virtual para apoio a uma Metodologia para Ensino de Algoritmos e Programação

José Carlos Rocha Pereira Júnior¹

Clevi Elena Rapkiewicz^{**}

Resumo. *Numa análise da literatura sobre o processo de ensino e aprendizagem de algoritmos e programação foram constatados melhores resultados agregando estratégias com ferramentas computacionais. Desta forma, este artigo propõe um ambiente virtual composto de ferramentas, que refletem uma metodologia, buscando amenizar os problemas deste processo.*

Palavras-chaves: *programação, metodologia de ensino, ambiente de aprendizagem.*

Abstract. *In analysis of literature about the teaching and learning process of algorithms and programming had been evidenced better resulted joining strategies with computational tools. In this way, this article considers a virtual environment composed by tools that reflects a methodology, searching to brighten up the problems of this process.*

Keywords: *programming, teaching methodology, learning environment.*

¹ Mestrando – Bolsista CAPES, Bacharel em Ciência da Computação, Universidade Estadual do Norte Fluminense – UENF, zeca@uenf.br

^{**} Professora/Pesquisadora, Doutora em Engenharia de Sistemas e Computação, Universidade Estadual do Norte Fluminense – UENF, clevi@uenf.br

1. Introdução

É uma atividade comum a resolução de problemas por parte dos profissionais da computação e engenharia. Tais cursos de formação por ventura promovem conteúdos associados ao desenvolvimento cognitivo para esta atividade requerida no cotidiano.

Este é o papel do aprendizado de programação em cursos de computação e engenharia, o que permite formalizar a forma de resolução de um determinado problema através do desenvolvimento de um programa para auxiliar estes profissionais em suas atividades.

Durante o processo de ensino-aprendizagem de fundamentos de programação nota-se que grande parte dos alunos apresenta dificuldades em assimilar as abstrações envolvidas. Não é novidade que disciplinas desta área apresentam altos índices de reprovação o que pode ser sintoma de limitadas pesquisas relacionadas (Kaasbøll, 1998). Segundo Pereira Júnior e Rapkiewicz (2004), a análise de 105 artigos sobre o tema mostra que há três vertentes na busca de soluções para os problemas do processo: *Ferramentas*, *Estratégias* e a união de ambas. Esta análise parece sugerir que a união de ferramentas computacionais e estratégias têm se demonstrado como melhor proposta. Porém, o que se observa é uma tendência em tratá-las separadamente.

A partir destas constatações propõe-se um ambiente de aprendizagem que reflita uma estratégia buscando a melhoria do processo em questão. Neste contexto, considera-se esta proposta aplicável ao ensino a distância e presencial. Destacamos que o ambiente não é simplesmente mais uma ferramenta computacional de apoio ao ensino de programação, mas fundamenta-se e guia uma estratégia baseada em resolução de problemas, o que representa seu diferencial.

2. Validação de uma Metodologia no Ensino Médio

A metodologia utilizada se baseia na estratégia descrita em Delgado *et al* (2004) que difere muito do método tradicional de ensinar tal conteúdo. Neste trabalho a estratégia é apoiada por um conjunto de ferramentas na tentativa de se obter melhores resultados, conforme predito na anteriormente.

Esta proposta se firma na tentativa de fazer com que os alunos utilizem o computador a maior parte do tempo, isto inclui a resolução de problemas através de jogos computacionais, interação por meio de um fórum de discussão e o desenvolvimento de programas por meio de uma linguagem de programação, pois representa a realidade de como é o trabalho profissional, aumentando a proficiência dos alunos e permitindo trabalhar, em parte, o inglês exacerbando a interdisciplinaridade que a programação poderá propor. No caso opta-se em utilizar a linguagem Pascal por ser a mais comum para esta aprendizagem no meio acadêmico e pela facilidade da tradução dos algoritmos, por sua proximidade com o português estruturado.

Na metodologia original a construção de algoritmos se dava em última instância e não constava a utilização de alguma linguagem de programação e, pelos motivos supracitados, decidiu-se agregar a tradução dos algoritmos para uma linguagem de programação.

É importante fazer os alunos entenderem todo o processo de aprendizagem a ser vivenciado para não gerar um descontentamento durante o seu desenvolvimento, pois a ânsia inicial se deve ao desejo da programação propriamente dita cuja atividade só será realizada na última etapa do processo.

Assim, considerando a união da estratégia com ferramentas, a metodologia é composta de três fases:

A primeira fase da estratégia visa desenvolver o raciocínio lógico e a criatividade dos alunos a partir da resolução de problemas que, segundo Falkembach *et al* (2003),

envolve a criação de algum procedimento para chegar à solução. São apresentados aos alunos alguns problemas que envolvem matemática básica e raciocínio lógico, passando a posterior resolução de jogos lógicos de computador.

A segunda fase busca a formalização de soluções onde, num primeiro momento, os alunos são motivados a propor suas soluções formalizadas em linguagem natural para serem avaliadas e discutidas pelos outros alunos através de um fórum de discussão. Num segundo momento, busca-se a formalização mais rígida e mais adequada pela questão fim, programar o computador. Neste momento, os alunos passam a formalizar buscando uma notação e passam a utilizar a estrutura de seleção “Se”.

Na terceira fase, a formalização dos algoritmos produzidos pelos alunos na segunda fase é traduzida para uma linguagem de programação, no caso Pascal. Nesta fase, “o aluno deve focalizar o processo de solução e não mais a solução em si” (Delgado *et al*, 2004). Assim, estes passam a reforçar e visualizar os resultados de suas propostas de soluções ao mesmo tempo em que passam a aprender uma linguagem de programação.

Para fazer uma avaliação da metodologia optou-se por adotar a proposta de inclusão de conteúdos de computação no ensino médio (formação geral) do Grupo de Trabalho em Licenciatura em Computação (GT3) da Sociedade Brasileira de Computação (SBC). Para tanto foi desenvolvida uma oficina de lógica de programação para projeção deste estudo de caso e definição de metas para o desenvolvimento do ambiente. A oficina foi realizada no Colégio Estadual Baltazar Carneiro em Cardoso Moreira – RJ com alunos do segundo ano e está descrita em detalhes no artigo de Pereira Júnior *et al* (2005).

O conteúdo de lógica está intrínseco na metodologia original, pois os alunos considerados são do ensino superior e já possuem tal conhecimento adquirido em uma disciplina a parte. Este não é o caso do ensino médio, onde se fez necessário explicitar tal conteúdo (conectivos lógicos: *e*, *ou* e *não*) tornando-o um complemento à oficina durante a segunda fase.

Durante a realização da primeira e segunda fase ocorreu elevado grau de interatividade principalmente durante a discussão das soluções propostas, na utilização dos jogos computacionais e no trabalho com o fórum. Esta abordagem de promover o trabalho coletivo aumentou o desejo pela resolução os problemas propostos, acarretando o desenvolvimento cognitivo quanto a este quesito sem a percepção dos alunos.

As dificuldades apresentadas pelos alunos não diferiram muito das apontadas na literatura e presentes no ensino superior. Destacam-se a dificuldade de descrever formalmente condições lógicas em uma estrutura de seleção, por exemplo; desenvolver programas que envolviam determinados conteúdos de matemática; e em se adaptarem a formalização rígida da linguagem de programação. O resultado de uma avaliação final mostra que dos 5 alunos concluintes, dos 16 inscritos, 4 possuem capacidade satisfatória de programar.

Nesse contexto, este artigo apresenta a arquitetura de um ambiente virtual para ser utilizada com a metodologia proposta de forma a haver integração entre estratégia pedagógica e ferramenta computacional, além de proporcionar aos alunos contato com o computador durante todo o processo.

3. A Arquitetura do Ambiente Virtual de Ensino de Programação – AVEP

O ambiente permite aplicar a metodologia utilizando um conjunto de ferramentas. Com isto, busca-se alcançar melhores resultados no aprendizado de algoritmos e programação, conforme já comentado em função da análise da literatura, integrando ferramenta e estratégia.

De forma geral, o ambiente é composto pelos Componentes de Aprendizagem (Fase 1, Fase 2 e Fase 3), pelo Componente de Comunicação, pelo Componente de Armazenagem e por um grupo de Atores (Aluno, Tutor e Administrador) que interagem.

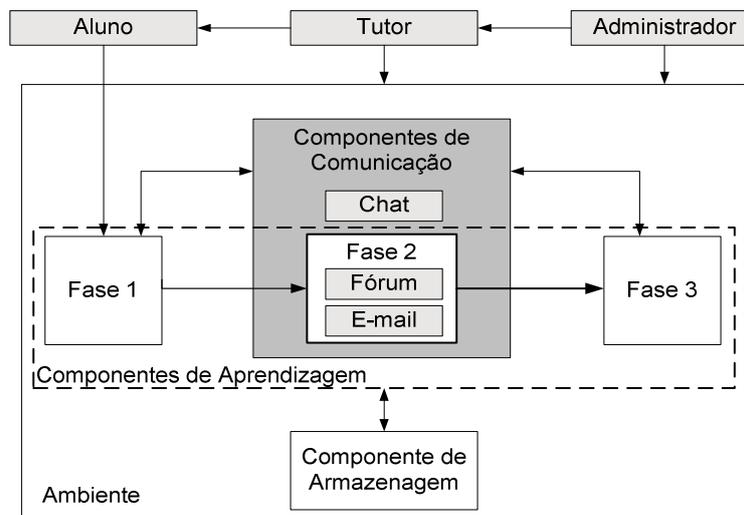


Figura 1. Arquitetura do AVEP

A aprendizagem irá fluir principalmente através da transferência entre os componentes de aprendizagem, ou seja, de uma fase para a outra segundo a abordagem pré-definida. Neste processo, o Componente de Comunicação está presente e contribui em todas as fases para que a aprendizagem seja mais eficaz e tenha aspecto colaborativo, pois permite que os atores se ajudem na partilha de questões, discernimentos e soluções; e passem a confiar uns nos outros para atingir o objetivo proposto.

Pode ser questionado aqui o uso do termo colaborativo ao invés de cooperativo. Os dois termos denotam a interação entre indivíduos, entretanto a cooperação é mais organizada, com enfoque no controle das informações, sendo mais utilizadas em ambientes computacionais.

Segundo Panitz (1996), a colaboração é uma filosofia de interação e de estilo de vida onde a cooperação é a estrutura da interação projetada para facilitar o acompanhamento da um produto final ou objetivo.

Apesar do aspecto cooperativo do ambiente proposto será insistido o tratamento colaborativo, pois se considera o desenvolvimento cognitivo do indivíduo a partir da livre troca de idéias para alcançar um mesmo objetivo, no caso, aprender a programar.

Dentro do contexto da Aprendizagem Colaborativa Assistida por Computador (Computer Supported Collaborative Learning – CSCL) que representa uma estratégia para construção do conhecimento mediado por recursos informáticos; e do Trabalho Cooperativo Apoiado por Computador (Computer-Supported Cooperative Work – CSCW) que busca descrever como as pessoas trabalham juntas com o computador, o ambiente em discussão está inserido de forma abstrata neste contexto na forma do chamado *groupware*, o conjunto das ferramentas de colaboração.

3.1. Atores

Um ator é considerado neste contexto como um componente que age ou interage sobre o ambiente. Esta atuação se dá de forma diferente segundo uma hierarquia na qual o nível mais alto é representado pelo *administrador*, o elemento mediador é o *tutor* e o nível mais baixo é ocupado pelo *aluno*. Para o administrador e o tutor não existem restrições para interagir sobre os componentes de aprendizagem, o que difere do aluno cuja interação é gradual, ou seja, ele inicia a aprendizagem a partir do primeiro componente

de aprendizagem e à medida que desenvolve as atividades daquele componente e adquire as competências necessárias consideradas pelo *tutor*, passa para o próximo componente e assim por diante até ter desenvolvido as competências correspondentes a conclusão da aprendizagem do conteúdo proposto.

3.1.1. Administrador

O *Administrador* é o ator que possui a maior gama de privilégios e sua atuação envolve principalmente a coordenação de tutores e a manutenção do ambiente. Como está no mais alto nível da hierarquia de atores, ao administrador são legados os privilégios dos atores dos níveis mais baixos.

3.1.2. Tutor

O *Tutor* é responsável por atribuir e organizar os elementos aos Componentes de Aprendizagem, por formar um grupo de alunos sob sua responsabilidade, por definir prazos para realização de atividades e por avaliar os resultados da aprendizagem de cada aluno. Assim, o papel de *tutor* é desempenhado pelo professor na forma de facilitador do processo de aprendizagem deixando o aluno ser responsável pelo próprio desenvolvimento cognitivo, característica de uma aprendizagem significativa.

3.1.3. Aluno

O *Aluno* é o principal ator do ambiente, pois se pretende com sua atuação sobre os Componentes de Aprendizagem o desenvolvimento de competências necessárias para a construção de programas de computador. A partir dos elementos do Componente de Comunicação disponíveis durante a aprendizagem o *aluno* poderá agregar maior conhecimento a partir de experiências cognitivas vivenciadas por outro *aluno* e a partir da comunicação com o *tutor*. No Componente de Armazenagem haverá conteúdo disponível ao *aluno* que também o auxiliará no processo.

É possível notar que apesar do ambiente de certa forma conduzir o processo de aprendizagem organizado pelo *tutor*, o aluno detém a responsabilidade agregar valor ao próprio conhecimento a partir de todo o ferramental disponível.

5.2. Componentes

Um componente é uma composição de elementos que possuem um objetivo comum. O ambiente apresenta três tipos de componente:

5.2.1. Componentes de Aprendizagem

Como o próprio nome refere, os Componentes de Aprendizagem são responsáveis por fornecer os elementos necessários para que ocorra a aprendizagem. Cada componente representa uma fase da metodologia considerada.

a) Fase 1

O objetivo deste componente é permitir ao Aluno trabalhar aspectos cognitivos da resolução de problemas. Este componente possui dois elementos: um questionário de problemas e um conjunto de jogos computacionais. O primeiro corresponde a uma lista de problemas envolvendo matemática básica e lógica selecionados e atribuídos pelo *tutor* cujas soluções serão avaliadas pelo mesmo; o segundo são jogos que envolvem raciocínio lógico dispostos num repositório, permitindo uma breve descrição da estratégia utilizada o que de forma imediata auxiliará o processo de aprendizagem proposto na próxima fase. Ao final o Aluno terá exercitado sua capacidade de resolver problemas.

Além dos alunos desenvolverem a cognição para resolução de alguns problemas a partir destes jogos, estes apresentam de forma intrínseca conteúdos teóricos que serão aproveitados nas outras fases. Como exemplo de jogo, propõe-se resolver o seguinte problema: “Dividir 8 litros de vinho em 2 porções de 4 litros manipulando apenas três garrafas de 8, 5 e 3 litros, respectivamente”. Desta forma, ao preencher uma garrafa com determinada capacidade de vinho é possível explicar o conceito de variável e de atribuição de valores.

b) Fase 2

O objetivo é capacitar o Aluno a descrever formalmente seu raciocínio. Este processo é dado de forma gradual, partindo da linguagem natural e finalizando com a construção de algoritmos em pseudo-código.

Este componente de aprendizagem compreende e se restringe a dois elementos do Componente de Comunicação: fórum e e-mail. Baseia-se na disponibilização de problemas através de um fórum no qual o Aluno irá propor sua solução a ser avaliada por outras instâncias de Aluno e pelo Tutor. É pretendido que toda a comunicação associada ao problema seja realizada através deste fórum para incentivar o aperfeiçoamento da descrição formal e correta de soluções. O e-mail neste caso é um elemento para comunicação extra, ou seja, não será considerado para a avaliação das soluções apresentadas e do aprimoramento da formalização requerida, seu uso está aberto a comunicação inter-pessoal para os mais variados assuntos.

c) Fase 3

Este componente visa o processo de tradução dos algoritmos desenvolvidos na Fase 2 para uma linguagem de programação, no caso, Pascal. Fornece recursos para a aprendizagem básica da linguagem como conteúdo didático e ambiente de construção, depuração e execução de programas.

3.2.2. Componente de Comunicação

Este componente permite a comunicação entre os atores do ambiente. É constituído de três elementos: chat, fórum e e-mail.

O *chat* é o elemento que permite a comunicação em tempo real entre os atores, o fórum é um espaço de discussão assíncrono e o *e-mail* permite a troca de mensagens através do correio eletrônico.

Este componente está sempre disponível no ambiente para qualquer ator. Entretanto, o elemento *chat* têm o acesso negado no está se atuando no Componente de Aprendizagem Fase 2 para induzir a utilização do fórum.

3.2.3. Componente de Armazenamento

Este componente é um repositório de dados para a inclusão de livros eletrônicos, apostilas e links que auxiliem o processo de aprendizagem, ou seja, qualquer material que agregue valor a aprendizagem. Está sempre disponível, porém a inclusão de conteúdo está atribuída ao Tutor e ao Administrador evitando assim que qualquer outro tipo de material que foge ao conteúdo abordado seja disponibilizado.

4. Considerações Finais: onde está e para onde vai

Destaca-se em vários trabalhos e relatos de professores e alunos que o processo de ensino e aprendizagem de algoritmos e programação apresenta um conjunto de dificuldades que tem origem em diversas vertentes e razões. Na análise da literatura

constata-se a existência na literatura de três vertentes: ferramentas, estratégias e a união de ambas. O que se averigua é a obtenção de melhores resultados com aplicação da última vertente, a união de ferramentas e estratégias, principalmente sob a forma de redução do índice de evasão e repetência.

A partir deste resultado foi proposta a arquitetura de um ambiente de aprendizagem que se fundamenta e guia uma estratégia, o que representa seu diferencial em relação a muitas outras ferramentas, uma vez que em grande parte visam somente à execução de programas.

O ambiente está sendo desenvolvido para ser utilizado na web. Assim, estará sempre disponível em qualquer lugar onde haja possibilidade de acesso a internet, além de considerar um custo mínimo de hardware e software para os padrões atuais.

As ferramentas de apoio a Fase 1 já foram utilizadas por uma turma de 27 alunos do curso de graduação em Engenharia de Produção da UENF, primeiro período. Uma análise preliminar dos resultados aponta para a motivação do grupo a desenvolver atividades coletivas e a motivação propiciada pelo contato inicial com computador já na resolução de problemas (fase lúdica). Há, porém a necessidade de ampliar o uso de jogos computacionais de maior complexidade.

Apesar da validação do ambiente realizada com alunos do ensino superior, nada será um empecilho para que alunos e não alunos de indeterminado nível de escolaridade tenham acesso e aprendam através do mesmo.

Referências

- Delgado, C., Xexeo, J. A. M., Souza, I. F., Campos, M., Rapkiewicz, C. E. (2004) “Uma Abordagem Pedagógica para a Iniciação ao Estudo de Algoritmos”. *XII Workshop de Educação em Computação*. WEI'2004, Salvador - BA.
- Falkembach, G. A. M., Amoretti, M. S. M., Tarouco, L. R., Viero, F. (2003) “Aprendizagem de Algoritmos: Uso da Estratégia Ascendente de Resolução de Problemas”. 8º Taller Internacional de Software Educativo. Santiago, Chile.
- Kaasbøll, J. (1998) “*Exploring Didactic Models for Programming*”. Norsk Informatikkonferanse, Høgskolen i Agder.
- Panitz, T. (1996) “*A Definition of Collaborative vs Cooperative Learning*”. Disponível em: <http://www.city.londonmet.ac.uk/deliberations/collab.learning/panitz2.html>. Última consulta em junho de 2005.
- Pereira Júnior, J. C. R., Rapkiewicz, C. E., Delgado, C., Xexeo, J. A. M. (2005) “Ensino de Algoritmos e Programação: Uma Experiência no Nível Médio”. In: *XIII Workshop sobre Educação em Computação*. WEI'05, São Leopoldo - RS.
- Pereira Júnior, J. C. R.; Rapkiewicz, C. E. (2004) “O Processo de Ensino e Aprendizagem de Algoritmos e Programação: Uma Visão Crítica da Literatura”. In: *III Workshop de Educação em Computação e Informática do Estado de Minas Gerais*, WEIMIG'04, Belo Horizonte - MG.