Uma Abordagem Metodológica voltada para o Ensino-Aprendizagem de Algoritmos

Marcelo Batista de Souza, Universidade Federal de Roraima, marcelolada@gmail.com

João Luis Gomes Moreira, Universidade Federal de Roraima, joaoluisgm@gmail.com

Felipe Leite Lobo, Universidade Federal de Roraima, guaxalobo@gmail.com

Márcio Aurélio dos Santos Alencar, Universidade do Estado do Amazonas, xmarcioaureliox@gmail.com

Resumo. *Dificuldade na aprendizagem de algoritmos é uma realidade enfrentada por estudantes de graduação da área de ciências exatas. O presente artigo investiga as causas do problema, descrevendo uma experiência que integrou ábaco, operações básicas da matemática e sistemas de numeração com o objetivo de explorar o raciocínio lógico dos estudantes e prepará-los para o estudo dos algoritmos. Foram utilizados na implementação Visualg e Free Pascal para verificar a influência do idioma do software na aprendizagem. O ambiente Moodle apoiou as discussões sobre o tema e os resultados da pesquisa apontaram avanços no desenvolvimento de algoritmos e programas de computador.*

Palavras-chave: *dificuldades de aprendizagem, sistemas de numeração, algoritmos.*

A Methodological Approach toward the Teaching-Learning Algorithm

Abstract. *A difficulty in learning algorithms is a reality faced by graduate students in the field of exact sciences. This paper investigates the causes of the problem, describing an experience that integrated abacus, basic operations of mathematics and numbering systems in order to explore the logical reasoning of students and prepare them for the study of algorithms. Were used in the implementation VisuAlg and Free Pascal to verify the influence of language learning software. The Moodle environment supported discussions on the subject and the results of the research show advances in the development of algorithms and computer programs.*

Keywords: *learning difficulties, number systems, algorithms.*

1. Introdução

Na *internet*, é possível encontrar vasta literatura sobre algoritmos e dificuldades de aprendizagem associados a estudantes da computação. Aureliano e Tedesco (2012) destacam que conceitos básicos de programação não estão sendo aprendidos, combinados e utilizados na prática. O problema tem sido enfrentado também por estudantes de outros cursos de graduação da área de ciências exatas. Diagnosticar as suas causas e apresentar propostas para minimizá-lo tem sido um papel do pesquisador.

Este trabalho foi baseado nas dificuldades de aprendizagem enfrentadas por estudantes no estudo de algoritmos. As disciplinas Introdução à Ciência da Computação (ICC) e Cálculo Numérico (CN) abordam o assunto. Ambas são ofertadas pelo Departamento de Matemática da Universidade Federal de Roraima (UFRR) para cursos de ciências exatas, diferentes da área da computação. Estudantes regularmente matriculados nas referidas disciplinas participaram da pesquisa. Foram investigadas as

possíveis causas relacionadas ao problema e utilizadas atividades didático-pedagógicas com o objetivo de explorar o raciocínio lógico dos estudantes e prepará-los para o estudo dos algoritmos. A influência do idioma do *software* na aprendizagem também foi verificada. Os resultados apontaram avanços no desenvolvimento de algoritmos. Na pesquisa, foram utilizados sala de aula e laboratório de informática do Centro de Ciências e Tecnologia da UFRR, que atende cursos de graduação e pós-graduação. Nas máquinas foram instalados os *softwares Free Pascal* (*FP*) e *Visualg*. O Ambiente Virtual de Aprendizagem (AVA) *Moodle* apoiou às discussões sobre o tema.

O restante do artigo está organizado e apresenta na seção seguinte os trabalhos correlatos que contribuíram para o desenvolvimento da pesquisa, destacando a utilização da metodologia na seção 3 e as ferramentas utilizadas na seção 4. As seções 5 e 6 descrevem os resultados alcançados e as conclusões do estudo, respectivamente. As referências que sustentam a discussão sobre o tema finalizam o trabalho.

2. Trabalhos Correlatos

Nesta seção, serão apresentados os trabalhos correlatos que destacam diferentes abordagens, adoção de estratégias e o desenvolvimento de ferramentas voltadas para o ensino. Os autores apresentaram suas contribuições como uma solução para as dificuldades de aprendizagem enfrentadas por estudantes no estudo de algoritmos e programação de computadores.

Gomes et al. (2008) apresentaram uma abordagem para ajudar estudantes com dificuldades de aprendizagem nos conceitos iniciais de programação. Atividades lúdicas foram utilizadas no sistema para desenvolver o raciocínio lógico, estimulando-os a resolver problemas propostos. O ambiente AVEP foi desenvolvido por Pereira Junior et al. (2006) que basearam o seu desenvolvimento na resolução de problemas. Eles sustentam que o uso computacional da ferramenta apóia o ensino de programação, mas não se limita apenas a execução de códigos.

Outro trabalho que auxilia na aprendizagem de programação é o de Hostins e Raabe (2007). Os autores descrevem o *WebPortugol* no qual é permitida a edição de algoritmos em *Portugol*. Resultados parciais apontaram evolução na construção do conhecimento e desenvolvimento da autonomia do estudante. Moreira e Favero (2009) implementaram um AVA que explora o ensino automático de programação. Nele, dois modelos de regressão linear medem a solução de algoritmos num processo iterativo/interativo, onde o estudante é orientado a refinar a sua própria solução.

Na seção seguinte será descrita a metodologia utilizada na pesquisa. Ela foi baseada em relatos de estudantes de turmas anteriores, matriculados nas disciplinas de ICC e CN, que apontaram dificuldades em desenvolver um raciocínio lógico que os permitissem solucionar computacionalmente problemas propostos. A abordagem considerou índices de evasão, reprovação e desmotivação.

3. Metodologia

No primeiro contato com os estudantes em sala de aula, eles ficaram surpresos quando perguntados se sabiam somar, subtrair, multiplicar e dividir números. As reações originaram a experiência relatada por Souza (2011) que integrou Operações Básicas da Matemática (OBM), Sistemas de Numeração (SN) e ábaco. O autor destaca que este foi o passo inicial para introduzir os conceitos de estrutura e funcionamento de um algoritmo.

3.1 O Sistema Decimal como Referência para Outros Sistemas

Souza (2011) explorou registros sobre a história e evolução das máquinas como os publicados por Fedeli et al. (2003). Ele destacou que desde 2000 a.C., o ábaco decimal já era utilizado pelos povos do Oriente em colheitas no campo e que a partir daí, surgiram máquinas com diferentes SNs. O autor utilizou o SN decimal como referência para outros SNs com o objetivo de familiarizar os estudantes a manipulação dos números e desenvolver neles, a capacidade de raciocínio lógico ao realizar operações matemáticas.

3.1.1 Conversões de Bases Numéricas

Souza (2011) destacou a importância das operações no SN posicional e utilizou um número decimal qualquer com *n* dígitos, para evidenciar a referência do sistema. Ele representou o número utilizando a notação $X = (d_n d_{n-1} d_{n-2} \ldots d_3 d_2 d_1)_{10}$, reescrito em seguida assim, $X = d_n .10^{n-1} + d_{n-1} .10^{n-2} + d_{n-2} .10^{n-3} + \ldots + d_3 .10^2 + d_2 .10^1 + d_1 .10^0$ ***(I).***

Analogamente, foi formalizada a escrita de qualquer número, independente da sua base, adotando como referência o SN decimal, ficando $Y = (d_n d_{n-1} d_{n-2} \ldots d_3 d_2 d_1)_\beta$, onde β representa a base numérica. Ou seja, partindo da idéia utilizada em ***(I)*** tem-se que $Y = d_n .\beta^{n-1} + d_{n-1} .\beta^{n-2} + d_{n-2} .\beta^{n-3} + \ldots + d_3 .\beta^2 + d_2 .\beta^1 + d_1 . \beta^0$. O recurso utilizado serviu para introduzir a Conversão de Bases Numéricas (CBN). A cada exposição sobre o assunto o grafo apresentado na Figura 1, ia sendo construído para mostrar as possibilidades de CBNs.

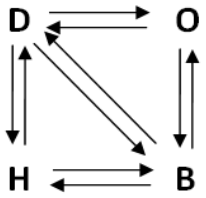

Figura 1 – Grafo representando as conversões possíveis entre as bases numéricas Decimal (D), Octal (O), Hexadecimal (H) e Binária (B).

Durante as aulas, também foram descritas as relações existentes entre as bases binária, octal e hexadecimal, representadas por 2^1, 2^3 e 2^4, respectivamente. Foram destacadas que cada dígito octal e hexadecimal é uma combinação de 3 e de 4 números binários. Ou seja, $X_1 = (d_n d_{n-1} d_{n-2} \ldots d_3 d_2 d_1)_8$ e $X_2 = (d_n d_{n-1} d_{n-2} \ldots d_3 d_2 d_1)_{16}$.

$d_1 d_2 d_3$ $d_1 d_2 d_3 d_4$

3.1.2 Operações Básicas Utilizando o Ábaco e Diferentes Sistemas de Numeração

Após a introdução dos SNs, as OBMs utilizando diferentes SNs, passaram a ser realizadas com o auxílio de um ábaco destacado por Silva e Salazar (2010) como uma ferramenta adequada para dar significado às técnicas operatórias dos algoritmos.

Foram apresentados vários exemplos para facilitar o entendimento dos estudantes utilizando a referência do SN decimal, conforme um deles pode ser visto na Figura 2. Nela, $s_1 = d_1 + d_1$, $s_2 = d_2 + d_2$, ..., $s_n = d_n + d_n$, apresentaria como resultado um número. Sendo qualquer um dos $s_n \geq \beta$, então o valor de β deveria ser trocado por unidades de uma casa de maior valor, o que dependeria da base utilizada na operação. Ou seja, considerando somente a casa das unidades e β =10, ter-se-ia $s_1 = d_1 + d_1 \geq 10x$

+ u_1, com $x \geq 0$ e $0 \leq u_1 \leq 9$, o resultado da soma proposta na Figura 2 ficaria assim $(s_n s_{n-1} s_{n-2} \ldots s_3[s_2+1]u_1)_{10}$.

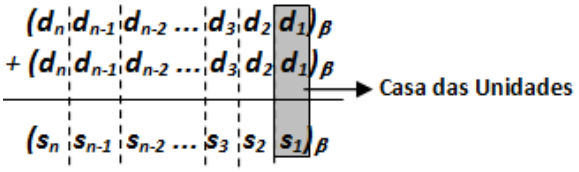

Figura 2 – Exemplo de operação básica da matemática.

Para outros casos em que $u_1 \geq 10$ bastaria observar a quantidade de unidades a ser trocada em cada uma das s_n. A idéia foi aplicada utilizando outras bases numéricas, a exemplo do trabalho de Souza (2011). Foram realizadas também operações de subtração, multiplicação e divisão com diferentes SNs.

3.2 Implementação Utilizando os *Softwares Free Pascal* e *Visualg*

Havia a necessidade de avançar sobre a estrutura e o funcionamento dos algoritmos. Foram propostos alguns problemas que proporcionaram discussões entre os estudantes sobre como resolvê-los. O entendimento do problema e a modelagem receberam ênfase, seguidos das definições de algoritmo, fluxograma e pseudocódigo. Na implementação, foram utilizados os *softwares FP* e *Visualg*, conforme dois exemplos podem ser vistos na Figura 3.

program soma; **var** A,B,S: **integer**; **begin** **write**('Digite o valor de A: '); **readln**(A); **write**('Digite o valor de B: '); **readln**(B); S:=A+B; **writeln**('O valor da soma é: '); **writeln**(S); **end**.	**algoritmo** "soma" **var** A,B,S: **inteiro** **inicio** **escreval** ("Digite o valor de A: ") **leia** (A) **escreval** ("Digite o valor de B: ") **leia** (B) S <- A + B **escreval** ("o valor da soma é: ") **escreva** (s) **fimalgoritmo**

Figura 3 – Programas desenvolvidos utilizando os *softwares Free Pascal* e *Visualg*.

Diferenças e semelhanças da sintaxe das linguagens também estão sendo destacadas na Figura 3. Após o desenvolvimento dos algoritmos, os estudantes realizaram os testes de mesa para verificar os resultados da execução. Segundo eles, a confiabilidade aumentou quando os algoritmos passaram a ser implementados na prática. A utilização dos *softwares* foi uma estratégia para avaliar a influência do idioma na aprendizagem do estudante. Na seção seguinte serão apresentadas as características das ferramentas utilizadas no experimento.

4. Ferramentas Utilizadas

Nesta seção, serão apresentadas as ferramentas computacionais utilizadas no desenvolvimento da pesquisa integrada pelos *softwares open source FP, Visualg e Moodle*. Todas estão disponíveis gratuitamente na *internet* para *download*.

4.1 O *Software Free Pascal*

O *FP* é um ambiente de desenvolvimento integrado, onde é possível editar e compilar programas de computador. Manzano e Yamatumi (2010) destacam que a ferramenta foi desenvolvida para o ensino de programação e acrescentam as vantagens dela ser estável,

multiplataforma e compatível com o formato usado pela *Borland*, empresa proprietária do *Turbo Pascal*.

Alguns trabalhos serviram de referência para que o *FP* e a linguagem *Pascal* fossem utilizadas na pesquisa, como o de Rocha et al. (2010) que adequou a linguagem no ensino inicial de programação imperativa, enfatizando a lógica de programação na resolução de problemas, o de Rocha e Ferreira (2011) que utilizou uma ferramenta de correção automática de algoritmos implementada no *FP*, aplicada a um sistema personalizado de ensino e o de Araújo (2006) que relatou experiências no aprendizado de algoritmos utilizando a linguagem *Pascal*, sendo apoiadas por um AVA.

4.2 O *Software Visualg*

O *Visualg* é um *software* que auxilia estudantes a escreverem algoritmos em português estruturado. Segundo Tavares et al. (2012) a ferramenta é simples e didática. Souza (2009) afirma que a experiência de professor o fez observar dificuldades na abstração, no desenvolvimento e na realização de testes de mesa. O autor argumenta que a idéia do *software* facilitaria a vida do iniciante em programação.

A escolha do *Visualg* foi influenciada pelos trabalhos de Barbosa et al. (2011) que a classifica como ferramenta de simulação, Anjos e Lucena (2012) que apresenta um programa baseado nas suas características e Dantas et al. (2012) que avalia o *software* no apoio à aprendizagem. Souza (2009) destaca que a exemplo do *Portugol*, linguagem de grande popularidade no meio acadêmico, era necessária a utilização de uma ferramenta na qual os princípios básicos da programação estruturada, pudessem ser ensinados sem comprometer a curva de aprendizagem.

4.3 O AVA *Moodle* como Ferramenta de Apoio

Moodle significa *Modular Object Oriented Dynamic Learning Enviroment.* Segundo Al-Ajlan e Zedan (2008), a ferramenta é baseada no construtivismo, voltada para o gerenciamento escolar. Souza et al. (2011) destaca algumas das suas vantagens como instalação, baixo custo, flexibilidade e facilidade na utilização, que o têm tornado uma opção para a criação de AVAs flexíveis e interativos.

A escolha do *Moodle* se baseou nos trabalhos de Aydin e Tirkes (2010) que destaca as vantagens dos seus recursos na *e-learning*, de Al-Ajlan e Zedan (2008) que apresenta um estudo comparativo com outros AVAs, de Alencar e Netto (2011) que combina o uso de tecnologias para melhorar a colaboração nos seus fóruns de discussão, e o de Kakasevski et al. (2008) que explora a questão da avaliação da sua usabilidade.

5. Resultados

A análise estatística dos dados se baseou na avaliação dos itens de um questionário misto, contendo perguntas objetivas e subjetivas. Participaram da pesquisa, duas turmas de 40 (quarenta) estudantes matriculados na disciplina de ICC.

A Figura 4 aponta a maior dificuldade enfrentada no estudo dos algoritmos e destaca se a atividade didático-pedagógica utilizando exercícios sobre SNs, OBMs, CBNs e ábaco, colaborou para o desenvolvimento do raciocínio lógico dos estudantes que responderam a uma pergunta objetiva.

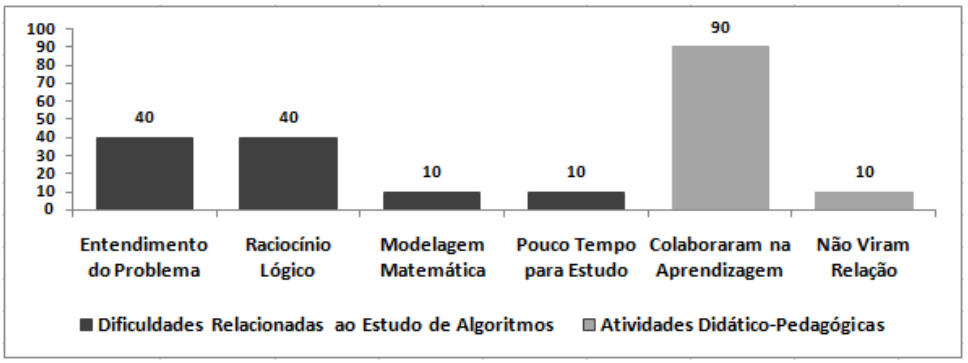

Figura 4 – Dificuldades e atividades práticas no estudo dos algoritmos.

Observa-se que o entendimento do problema e o raciocínio lógico compuseram a maioria de 80% das respostas apresentadas na Figura 4. Todavia, a modelagem matemática e a dedicação aos estudos também foram citadas. As respostas dadas sugerem a relação do baixo rendimento com a falta de amadurecimento do estudante. Os entrevistados que não viram relação das atividades didático-pedagógicas com a aprendizagem de algoritmos, alegam que os conteúdos da disciplina foram passados rapidamente. Isto não reflete algo incomum dentre as reclamações dos calouros, pois ainda estão se adaptando à nova rotina. A maioria dos entrevistados que não tiveram dificuldades na disciplina aprovaram a metodologia.

A Figura 5 apresenta se houve influência do uso do *software FP* na aprendizagem dos estudantes, onde os entrevistados, que responderam a uma pergunta subjetiva, ficaram divididos em 50%.

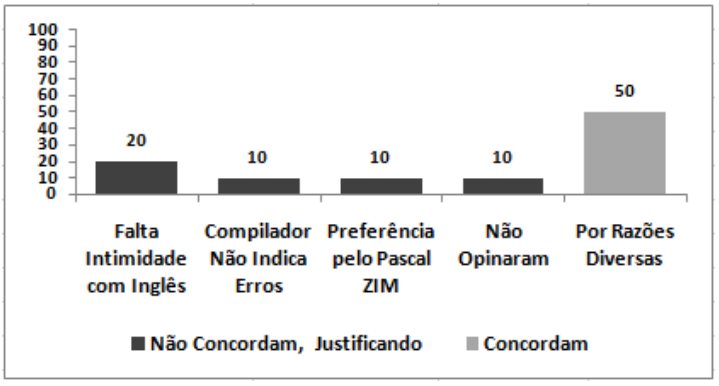

Figura 5 – Colaboração do *software Free Pascal* na aprendizagem de algoritmos.

As barras escuras da Figura 5 indicam algumas razões expostas pelo grupo. A questão do idioma do *software* e a busca por encontrar erros em códigos se tornarão triviais com a convivência no meio acadêmico e a dedicação aos estudos. Quanto à preferência pelo *Pascal Zim*, é aceitável a sua utilização no início de ICC, porém como não dispõe de estruturas de dados trabalhadas desde os algoritmos até o uso da linguagem, não é recomendada para a disciplina. A Figura 5 destaca na barra clara os 60% dos entrevistados que apontaram a colaboração do *FP* na aprendizagem. As razões foram diversas como a oportunidade de exercitar a língua inglesa, utilização dos seus recursos no futuro, verificação do funcionamento e simples implementação do programa.

A Figura 6 apresenta a opinião dos estudantes quando perguntados sobre a influência do *software Visualg* na aprendizagem de algoritmos. As respostas apresentadas foram subjetivas. Neste quesito, o total de respostas dadas pelos

entrevistados empatou em 50% para aqueles que discordam e 50% para os que concordam. Do último grupo, apenas 10% afirmaram terem sido beneficiados e outros 90% não responderam ao questionamento, o que sugere uma nova coleta de dados.

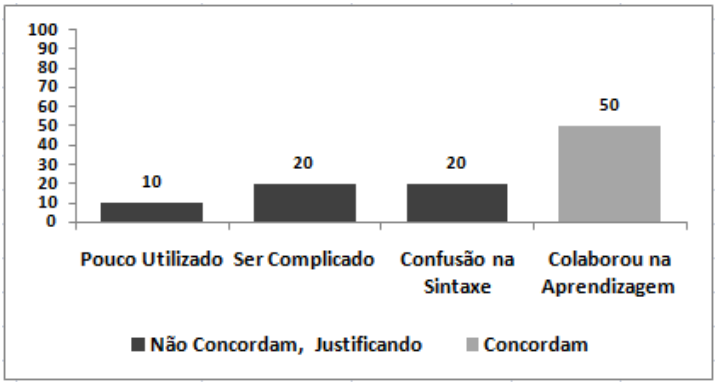

Figura 6 – Colaboração do s*oftware Visualg* na aprendizagem de algoritmos.

Para os entrevistados que discordam, alegando que o *software* foi pouco utilizado, é complicado e se confundem com a sintaxe, a baixa freqüência nas aulas de laboratório comprometeu a aprendizagem deles. Os demais destacam que o idioma ajudou no desenvolvimento dos algoritmos.

A Figura 7 mostra a relação das dificuldades dos estudantes no processo de aprendizagem de algoritmos com os idiomas do *FP* e *Visualg*, além da preferência do grupo por utilizar um dos dois *softwares* na implementação. Os entrevistados responderam a uma questão subjetiva.

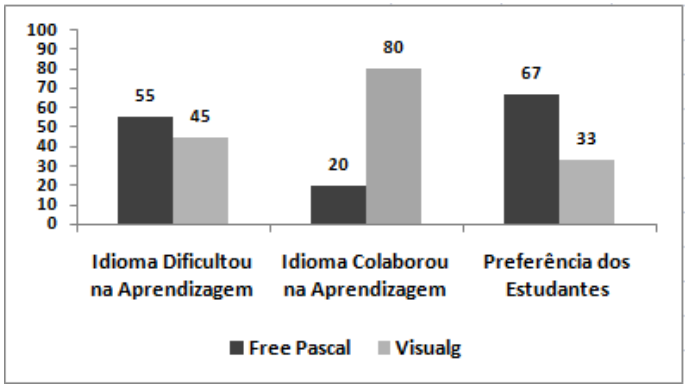

Figura 7 – Influência do idioma na aprendizagem e escolha do compilador.

Na Figura 7, 55% dos entrevistados disseram que as dificuldades de aprendizagem estão relacionadas ao idioma do *FP*. Outros 45% foram contrários a essa opinião. A maioria justificou que as informações em inglês atrapalham quando o código apresenta erros. Por outro lado, a minoria preferiu trabalhar com o *FP*, porque considerou a sintaxe mais fácil, similar a do *Visualg* e colaborou para a superação de dificuldades. Contudo, neste quesito as estatísticas carregam a subjetividade dos entrevistados quanto ao interesse e conhecimento prévio sobre a língua inglesa.

Foi perguntado também se a utilização do *software Visualg*, em português, colaborou para a aprendizagem de algoritmos, ao passo que 80% deram respostas positivas e 20% negativas com justificativas pessoais que nada têm a ver com o idioma. As alegações para as afirmações positivas apontam que 40% se referem ao idioma como efetiva colaboração no estudo, 10% reportaram a linguagem do *Visualg*, 10% relataram a vantagem de indicação de erros, 10% informaram que o idioma pouco ajuda no

processo e 10% responderam que facilita no desenvolvimento de algoritmos. Para finalizar a pesquisa os entrevistados emitiram opinião, destacada na Figura 7, sobre qual dos dois *softwares* acharam melhor para implementar os algoritmos. O *FP* foi a opção apresentada pela maioria com 55% das respostas, o *Visualg* com 28%, o *Pascal Zim* foi sugerido por 8,5% e o restante de 8,5% não respondeu a pergunta. Na seção seguinte serão apresentadas as conclusões da pesquisa e os trabalhos em andamento.

6. Conclusões e Trabalhos em Andamento

Neste trabalho foi apresentada uma abordagem que explorou o ensino de algoritmos utilizando ábaco, SNs e os *softwares FP* e *Visualg*. O foco da estratégia visou desenvolver nos estudantes raciocínio lógico, concentração e organização das idéias, para introduzir o estudo dos algoritmos.

Após o levantamento sobre o estado da arte, foi definida a integração de tecnologias baseado em diferentes abordagens. A pesquisa foi dividida em duas partes. A primeira teve relação com o uso de material concreto. A manipulação do ábaco utilizando a referência do SN decimal permitiu que as OBMs fossem realizadas com diferentes SNs para passar ao estudante a idéia do que é um algoritmo. A segunda refere-se à adoção e uso de *softwares open source* disponíveis gratuitamente na *internet*. As funcionalidades do *FP*, *Visualg* e *Moodle* não foram incluídas na pesquisa, visto que já existe na literatura exposição sobre elas. As perguntas do questionário foram direcionadas ao estudante e exploraram basicamente a dificuldade no estudo dos algoritmos, atividades didático-pedagógicas e influência do idioma dos *softwares* na aprendizagem.

Os resultados da pesquisa sugerem um repensar sobre a oferta das disciplinas ICC para calouros e CN para veteranos. De qualquer forma, as dificuldades dos calouros não refletem algo incomum, pois estudantes de computação apresentam o mesmo problema. E mais, foi observado que o idioma inglês, geralmente nativo das linguagens de programação, não é o problema chave no estudo de algoritmos. As maiores dificuldades apontadas pelos estudantes foram o entendimento do problema e a capacidade de raciocinar logicamente. Ambos entraves nada mais são que processos existentes no estudo de qualquer área do conhecimento. Todavia, calouros no primeiro contato com disciplinas básicas de seus cursos, iniciam vivências universitárias pessoais, passando por atribuições pertinentes a inserção acadêmica, onde aprendem a formalizar o conhecimento e superar as dificuldades naturais.

Para finalizar, o que justifica a abordagem proposta no artigo foram os índices registrados em semestres anteriores de evasão, reprovação e desmotivação de estudantes nas referidas disciplinas. Na tentativa de combater o problema citado em outros contextos educacionais, alguns pesquisadores desenvolveram novas ferramentas a exemplo de Hostins e Raabe (2007), Moreira e Favero (2009) e Barbosa et al. (2011). Diferentemente deles, neste trabalho se tentou diagnosticar o problema enfrentado por estudantes, de outros cursos diferentes da área da computação. Para minimizá-lo foram utilizadas atividades didático-pedagógicas. Além disso, foram testados e analisados dois *softwares* com o objetivo de verificar se o idioma influencia na aprendizagem. Esta abordagem pretendeu desenvolver nos estudantes as habilidades necessárias para que tenham êxito no desenvolvimento de algoritmos utilizando técnicas de programação. Para trabalhos futuros serão incorporados os que já estão em andamento, dar continuidade ao estudo com o *software* Visualg, por conta da ausência de respostas apresentadas na Figura 6, utilizar jogos lógicos para desenvolver o raciocínio dos

estudantes e explorar o *software* Geany que é um editor de texto multiplataforma voltado para o desenvolvimento de algoritmos e programas de computador.

Este trabalho tem se desenvolvido dentro do Grupo de Pesquisa em Ensino a Distância (GPED/UFRR), registrado na Pró-Reitoria de Pesquisa e Pós-Graduação (PRPPG/UFRR) com apoio do Núcleo de Educação a Distância (NEaD/UFRR).

Referências Bibliográficas

ALENCAR, M. A. S.; NETTO, J. F. M. Improving cooperation in virtual learning environments using multi-agent systems and AIML. In: **FRONTIERS IN EDUCATION CONFERENCE**, 41., 2011, Rapid City. Anais. Rapid City: IEEE Frontiers in Education Conference. p. 1-6.

ALVES, F. T. O.; ALVES, C. A.; SILVA JUNIOR, L. M.; SILVA, S. V. P. A utilização do ábaco no processo de ensino e aprendizagem do sistema de numeração decimal e dos algoritmos de adição e subtração nos anos iniciais. In: **ENCONTRO PARAÍBANO DE EDUCAÇÃO MATEMÁTICA**, 6., 2010, Monteiro. Anais. Monteiro: Encontro Paraíbano de Educação Matemática.

ANJOS, A. O.; LUCENA, L. Definição da sintaxe abstrata de potigol em XML. In: **CONGRESSO NORTE NORDESTE DE PESQUISA E INOVAÇÃO**, 7. 2012, Palmas. Anais. Palmas: Congresso Norte Nordeste de Pesquisa e Inovação.

ANJOS, M. L.; PIRES, M. N. M. Sistema de numeração decimal e os algoritmos das operações fundamentais. **Dia a Dia Educação – Portal Educacional do Estado do Paraná, Curitiba**. p. 1847-1848, 2009.

BARBOSA, S. L.; FERNANDES, T. C. B.; CAMPOS, A. M. C. Takkou: uma ferramenta proposta ao ensino de algoritmos. In: **WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO**, 18., 2011, Natal. Anais. Natal: Congresso da Sociedade Brasileira de Computação.

DANTAS, S. S.; LOPES FILHO, J. G.; MORAIS, L. A. M.; RODRIGUES, R. S.; COSTA, R. A. Avaliação de um software educacional de apoio à aprendizagem de programação: VisuAlg. In: **CONGRESSO NORTE NORDESTE DE PESQUISA E INOVAÇÃO**, 7., 2012, Palmas. Anais. Palmas: Congresso Norte Nordeste de Pesquisa e Inovação.

FEDELI, R. D.; POLLONI, E. G. F.; PERES, F. E. **Introdução à ciência da computação.** São Paulo: Ed. Pioneira Thomson Learning, 2003. 158 p.

GOMES, A.; HENRIQUES, J.; MENDES, A. J. Uma proposta para ajudar alunos com dificuldades na aprendizagem inicial de programação de computadores. **Journal Educação, Formação e Tecnologias**. Coimbra. Maio 2008, vol.1 (a), p. 93-103 Disponível em http://eft.educom.pt.

HOSTINS, H.; RAABE, A. Auxiliando a aprendizagem de algoritmos com a ferramenta Webportugol. In: **WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO**, 14., 2007, Rio de Janeiro. Anais. Congresso da Sociedade Brasileira de Computação.

MANZANO, J. A. N. G.; YAMATUMI, W. Y. **Free Pascal: programação de computadores.** São Paulo: Ed. Érica, 2010. 388 p.

MOREIRA, M. P.; FAVERO, E. L. Um ambiente para ensino de programação com feedback automático de exercícios. In: **WORKSHOP SOBRE EDUCAÇÃO EM**

COMPUTAÇÃO, 16., 2009, Bento Gonçalves. Anais. Congresso da Sociedade Brasileira de Computação.

PEREIRA JÚNIOR, J.C.R.; RAPKIEWICZ, C. E.; XEXEO, J.A.M.; DELGADO, C. AVEP – Um ambiente de apoio ao ensino de algoritmos e programação. In: **WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO**, 13., 2006, Campo Grande. Anais. Congresso da Sociedade Brasileira de Computação.

SANTOS, R. P.; COSTA, H. A. X. Análise de metodologias e ambientes de ensino para algoritmos, estruturas de dados e programação aos iniciantes em computação. **Infocomp – Journal of Computer Science**. Lavras. 2006, v. 5, nº 1., p. 41-50. Disponível em http://www.cos.ufrj.br/~rps/pub/periodicos/2006/INFOCOMP.pdf

SILVA, J. B. R.; SALAZAR, J. V. F. O ábaco na (re)significação do sistema de numeral decimal e das operações de adição e subtração com reagrupamento. In: **ENCONTRO PARAÍBANO DE EDUCAÇÃO MATEMÁTICA**, 6., 2010, Monteiro. Anais. Monteiro: Encontro Paraíbano de Educação Matemática.

SOUZA, C. M. VisuAlg - Ferramenta de apoio ao ensino de programação. **Revista TECCEN**, Vassouras, nº 2, 2009.

SOUZA, M. B. Integrando ábaco, sistemas de numeração e algotimos: relatos de uma experiência. In: **SEMANA DE MATEMÁTICA DA UNIVERSIDADE FEDERAL DE RORAIMA**, 5., 2011, Boa Vista.

SOUZA, M. B.; NETTO, J. F. M.; ALENCAR, M. A. S.; SILVA, M. M. Arcabouço telerobótico educacional baseado em sistemas multiagente. In: **SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO**, 23.; **WORKSHOP INFORMÁTICA NA ESCOLA**, 17.; 2011, Aracaju. Anais. Aracaju: Simpósio Brasileiro de Informática na Educação; Workshop Informática na Escola.

TAVARES, L. G.; CARSTENS, L.; DESCHAMPS, A. C. F.; FANCHIN, B.; FERREIRA, E. B.; CARMO, J. S. A importância das disciplinas de algoritmos e programação no desenvolvimento dos trabalhos de conclusão de curso na engenharia elétrica da universidade positivo. In: **CONGRESSO BRASILEIRO DE EDUCAÇÃO EM ENGENHARIA**, 40., 2012, Belém. Anais. Belém: Congresso Brasileiro de Educação em Engenharia.

TEDESCO, P. C. A. R.; AURELIANO, V. C. O. Ensino-aprendizagem de programação para iniciantes: uma revisão sistemática da literatura focada no SBIE e WIE. In: **SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO**, 23., 2012, Rio de Janeiro. Anais. Rio de Janeiro: Congresso Brasileiro de Informática na Educação.