

# Tepequém: uma nova Ferramenta para o Ensino de Algoritmos nos Cursos Superiores em Computação

Ornelio Hinterholz Jr.<sup>1</sup>

<sup>1</sup>Coordenação de Sistemas de Informação e Licenciatura em Computação – Faculdade Atual da Amazônia (FAA) – Boa Vista, RR - Brazil

ornelio@faculdadeatual.edu.br

***Abstract.** This paper presents a tool named Tepequém whose goal is to improve the learning process for beginners in Algorithm and Programming subject, which is problematic in computing undergraduate courses in Brazil due to high dropout and repetition. Still in validation process, this tool is actually a programming language whose main characteristic is to have all its keywords written in Portuguese only, allowing the students to write their programs using the mother tongue and eliminating the initial barrier of learning algorithms that is the English language.*

***Resumo.** Este artigo apresenta uma ferramenta denominada Tepequém que visa melhorar a aprendizagem dos alunos iniciantes no conteúdo de Algoritmos e Programação, o qual é problemático nos cursos superiores de computação no Brasil, devido ao alto índice de evasão e repetência. Ainda em fase de validação, tal ferramenta é na verdade uma Linguagem de Programação cuja principal característica é ter todas as suas palavras-reservadas escritas em português, possibilitando dessa maneira que o aluno possa criar seus programas utilizando a língua materna, eliminando assim o entrave inicial do aprendizado de algoritmos que é o idioma inglês.*

## 1. Introdução

Com a evolução da computação durante a segunda metade do século XX, foi permitida a criação de várias outras tecnologias que tornaram o mundo de hoje possível. De fato, as Novas Tecnologias da Informação e Comunicação (NTICs) geraram e continuam gerando impactos profundos sobre os indivíduos e a sociedade.

Diante deste contexto, temos que a disciplina que abarca os conteúdos de Algoritmos e Lógica de Programação é extremamente importante dentro das Matrizes Curriculares dos cursos de graduação em computação. Porém, conforme Rocha (1993) tal disciplina é uma das disciplinas que apresentam as maiores taxas de reprovação nos cursos de graduação do país (cerca de 60%).

Isso só demonstra que os pesquisadores da área da educação precisam oferecer propostas viáveis de maneira a diminuir tanto a evasão quanto a repetência nessa disciplina, possibilitando também uma formação de melhor qualidade para os egressos dos cursos de computação em nível de graduação.

## 2. O Ensino de Algoritmos

Segundo Rapkiewicz et al. (2006), a disciplina de algoritmos é considerada a base para o ensino de programação nos cursos de Ciência da Computação. Esta disciplina aborda

os princípios da lógica de programação, com o objetivo de desenvolver a capacidade de análise e resolução de problemas dos alunos através da descrição dos mesmos na forma de algoritmos.

Além disso, ela faz parte do plano curricular de outros cursos, como os de engenharia, com a finalidade de introduzir os conceitos de programação. Essas disciplinas costumam ter altos índices de evasão e reprovação, sendo um dos gargalos existentes nos cursos de graduação, particularmente de computação, dificultando ou até mesmo impedindo a continuidade dos alunos no curso.

Diversos problemas podem ser citados no processo de ensino e aprendizagem de algoritmos e programação que culminam em desmotivação, reprovação ou evasão. Entre esses, mencionados por Rodrigues (2002), pode-se destacar: dificuldade de adaptação dos alunos desenvolverem raciocínio lógico quando estão acostumados a decorar o conteúdo; falta de motivação do aluno gerada pelo despreparo e o desânimo quando ele acredita que a disciplina constitui um obstáculo extremamente difícil de ser superado.

Além desses motivos, pode ser apontado como fator limitante o ensino instrucionista: segundo Borges (2000), o modo tradicional não consegue facilmente motivar os alunos a se interessar pela disciplina. Entre outras razões, isso ocorre, pois não é clara para os mesmos a importância de certos conteúdos para sua formação.

Outro problema importante, apontado por Pereira e Rapkiewicz (2004), refere-se às dificuldades apresentadas por grande parte dos alunos em assimilar as abstrações envolvidas no processo de ensino e aprendizagem de programação. Além disso, muitos dos alunos não conseguem desenvolver o raciocínio lógico necessário para o posterior desenvolvimento de programas.

Analisando estas questões sob a ótica de uma pedagogia que encara a aprendizagem como um processo de construção de conhecimento, é possível identificar e intervir, por meio de estratégias adequadas, em diferentes etapas deste processo.

### **3. A Linguagem de Programação Tepequém**

Haetinger e Graziola (2002) nos dizem que “tornar o processo ensino-aprendizagem mais interessante é um desafio enfrentado pelo professor em qualquer nível de ensino”.

A utilização de novas ferramentas normalmente tendem a fazer com que as aulas tornem-se mais atrativas, participativas e o conteúdo seja assimilado pelos alunos de forma mais prazerosa, pois os mesmos sentem-se motivados a aprender e ansiosos pelos novos conhecimentos que os possibilitarão resolver problemas mais complexos criados por eles mesmos.

O Tepequém, ou “Tupã queem” que na língua Pemón significa “Deus do Fogo”, é uma Linguagem de Programação imperativa focada no paradigma estruturado, já que a mesma, mesmo podendo ser utilizada em semestres mais avançados, somente tem a finalidade de ser utilizada nos primeiros semestres dos cursos de graduação em computação, assim justificando a escolha do paradigma, caso contrário optaríamos por uma linguagem com um nível maior de abstração e de maior aceitação pela comunidade acadêmica e o mercado, neste caso, as linguagens que se utilizam do paradigma de orientação a objetos.

A principal característica da linguagem reside no fato de que suas palavras

reservadas para a construção básica de programas são escritas em sua totalidade em português, preservando inclusive a acentuação típica do idioma, como pode ser observado na figura 1. Tal escolha deve-se ao fato de que para os alunos que saem da Educação Básica, o idioma inglês, geralmente, é um obstáculo e prejudica o bom andamento do processo de ensino-aprendizagem no desenvolvimento dos conteúdos de algoritmos e programação.

## **Estrutura Geral de um Algoritmo**

**#não usa gráfico**

algoritmo **AtualTec**;

variáveis globais

```
{
    inteiro valor01;
    inteiro valor02;
    inteiro soma;
}
```

▪  
▪  
▪

## **EXEMPLO**

programa principal

```
{
    imprima("Digite o primeiro valor: ");
    valor01 = leialnteiro();
    imprima("Digite o segundo valor: ");
    valor02 = leialnteiro();
    soma = (valor01 + valor02);
    imprimaNL("A soma dos valores é: ", soma);
}
```

**Figura 1. Exemplo de código escrito em Tepequém**

A grande diferença do Tepequém para as outras linguagens do tipo “pseudo-código” é que a mesma é extensível, não limitando o aluno a utilização somente das funções básicas implementadas no pacote padrão. Isso acontece porque o código Tepequém não é compilado, mas traduzido primeiramente para a linguagem de programação Java e, somente, então, compilado para poder ser posteriormente interpretado em tempo de execução por uma Máquina Virtual. Dessa maneira, o Tepequém possui total integração com a Linguagem de Programação Java, herdando também suas características benéficas.

Outra característica que vale ser ressaltada é que o Tepequém é uma linguagem fortemente tipada e possui estruturação de código restritiva. Todas as restrições impostas foram implementadas para que o aluno se volte para questões mais importantes como: aprender as estruturas algorítmicas e desenvolver o seu raciocínio lógico, ao invés de perder tempo com minúcias que posteriormente podem facilmente ser assimiladas.

Por fim a linguagem Tepequém, apesar de poder ser trabalhada simplesmente utilizando um bloco de notas, foi integrada ao IDE (*Interactive Development Environment*) NetBeans 6.5 da Sun Microsystems, sob a forma de um *plug-in*, de maneira a poder se utilizar das funcionalidades extras que o mesmo oferece principalmente no que diz respeito a GUI (*Grafical User Interface*) do mesmo que é extremamente intuitiva e os Analisadores Léxico e Sintático, além do Colorizador de Código. Tudo isso possibilita ao aluno que durante sua atividade de programação o mesmo já saiba o que está errado com seu código, podendo corrigi-lo mesmo em tempo

de projeto.

Inicialmente em sua versão alfa, o Tepequém suporta somente aplicações em modo texto, porém já está sendo desenvolvida uma extensão que possibilitará aos alunos a utilização e criação de GUIs. O Plugin pode ser baixado em <http://plugins.netbeans.org/PluginPortal/faces/PluginDetailPage.jsp?pluginid=16632>.

#### **4. Considerações Finais**

Este trabalho teve a intenção de apresentar de maneira sucinta a problemática vivida no ensino superior de computação no que se refere ao desenvolvimento do processo de ensino-aprendizagem de Algoritmos e Programação e apresentar uma ferramenta que pode prover melhorias neste processo de forma a garantir uma melhor compreensão dos alunos, em relação a estes conteúdos, pelo fato de que a ferramenta estimula a prática e não possui entraves básicos na sua utilização como, por exemplo, o idioma estrangeiro.

Vale ressaltar que a metodologia adotada por cada docente para possibilitar aos alunos o conhecimento e desenvolvimento do raciocínio lógico, além da análise e solução de problemas, pode facilmente integrar a ferramenta de acordo com as necessidades levantadas no cotidiano de sala de aula.

O fato que segue é que estamos em uma nova era, com novos princípios, em uma dinâmica rápida e constante. Não podemos mais fechar os olhos para este novo mundo e queremos ensinar nossos alunos da forma como nós fomos ensinados, pois a sociedade e o mundo estão muito diferentes.

Devemos, então, pensar em formas novas de transmitir e fazer com que os conhecimentos cheguem até os nossos alunos, formas novas de aprendizagem que podem ser auxiliadas pelas tão temidas TICs, na visão de alguns educadores.

#### **Referências**

- Borges, M. (2000) “Avaliação de uma metodologia alternativa para a aprendizagem de programação”, <http://200.169.53.89/download/CD%20congressos/2000/SBC%202000/eventos/wei/wei006.pdf>. Acessado em: Julho 2008.
- Haetinger, W., Graziola, P. (2002) “Utilização de um carro-robô construído com componentes de baixo custo como ferramenta de apoio ao aprendizado dos conceitos de orientação a objetos”. 7º Taller Internacional de Software Educativo, <http://dialnet.unirioja.es/servlet/articulo?codigo=1233070>. Acessado em: Julho 2008.
- Pereira, J., Rapkiewicz, C. (2004) “O Processo de Ensino-Aprendizagem de Fundamentos de Programação: Uma Visão Crítica da Pesquisa no Brasil”, <https://www.sbc.org.br/bibliotecadigital/download.php?paper=44>. Acessado em: Julho 2008.
- Rapkiewicz, C. et al. (2006) “Estratégias Pedagógicas no Ensino de Algoritmos e Programação associadas ao uso de Jogos Educacionais”, CINTED-UFRGS - Novas Tecnologias na Educação. V.4 Nº 2, Dezembro, 2006.
- Rocha, H. (1993) “Representações Computacionais Auxiliares ao Entendimento de Conceitos de Programação”, In: “Computadores e Conhecimento: Repensando a Educação”. Livro organizado por Valente, J. A. Editora Unicamp.
- Rodrigues Jr, M. (2002) “Como Ensinar Programação?”, Informática – Boletim Informativo, Ano I, nº 01, ULBRA, Canoas, RS.