

## Utilização de Objetos de Aprendizagem em Sistemas Tutores Inteligentes para o ensino da Programação

Elvis Lopes de França<sup>1</sup>, Zildomar Carlos Felix<sup>2</sup>, Marcia da Silva Souza<sup>1</sup>, Thiago Brito Carneiro<sup>1</sup>, Paulo Roberto Cardoso Sousa<sup>1</sup>, Carlos Alberto Pereira Duarte Filho<sup>1</sup>

<sup>1</sup>Faculdade de Computação – Universidade Federal do Pará (UFPA)  
FACOM Campus de Marabá –68501-970 – Marabá – PA

<sup>2</sup>Universidade Federal Rural de Pernambuco (UFRPE)  
UAST – Unidade Acadêmica de Serra Talhada – Serra Talhada – PE

elvislf5@gmail.com, zildomar@ufrpe.br, macinhar-fofinha@hotmail.com,  
thiagobritocarneiro@hotmail.com, paulo\_beto\_cs@hotmail.com,  
carlinhosalbertoo@gmail.com

### RESUMO

*Alunos dos cursos de computação têm muitas dificuldades de aprendizagem nas disciplinas de programação. Nesse contexto o objetivo desse trabalho é apresentar os principais conceitos relacionados a utilização de Objetos de Aprendizagem nos Sistemas Tutores Inteligentes para o auxílio do ensino da programação, descrevendo as características, exemplos e as vantagens da utilização desses objetos nos STI. No artigo também é apresentada a proposta de criação de objetos com temas voltados a região amazônica, como forma de estimular a aprendizagem da programação de forma lúdica por alunos dessa região.*

Palavras-Chave: Objetos de Aprendizagem; Sistemas Tutores Inteligentes; Ensino de Programação.

### 1. INTRODUÇÃO

Atualmente existem diversas maneiras utilizadas para facilitar o ensino-aprendizagem nos ambientes educacionais. É o que chamamos de técnicas de aprendizagem, o que já é comum em várias escolas, universidades, centros de estudos, etc. Na área acadêmica da computação, é comum programadores iniciantes mostrar dificuldades nas disciplinas introdutórias, como lógica de programação e algoritmos sendo isso um obstáculo a ser superado.

Nesse contexto a utilização de novas técnicas de aprendizado torna-se essencial, dentre os vários recursos utilizados em outras áreas da educação pode-se destacar os Objetos de Aprendizagem (OA), que surge como possibilidade de potencializar o processo de ensino aprendizagem (JESUS et al, 2007). Além dos objetos pode-se destacar também os Sistemas Tutores Inteligentes que já são utilizados em diversas instituições para colaborar com o processo de ensino.

Este projeto objetiva demonstrar e implementar as formas de utilização dos OA em sistemas tutores inteligentes voltados para o ensino da programação, como forma de se produzir ferramentas mais flexíveis e que possam ser adaptadas para diferentes formatos de ensino e de aprendizagem. Pois esses objetos se caracterizam por serem desenvolvidos através dos conceitos da orientação a objetos (WILEY 2001).

O artigo está organizado nas seguintes seções: a seção 2 faz uma explanação sobre os objetos de aprendizagem; a seção 3 aborda as dificuldades do ensino-aprendizagem dos fundamentos da programação; a seção 4 apresenta a sugestão de utilização de Objetos de Aprendizagem em ambientes de ensino da programação.

## **2. OBJETOS DE APRENDIZAGEM**

Contemporaneamente professores tem adotado a utilização de Objetos de Aprendizagem no processo de ensino/aprendizagem, porém a sua definição ainda pode ser considerado vago, não existe um conceito que seja universalmente aceito e muitos outros termos são utilizados. Isto sempre resulta em confusão e dificuldade de comunicação, o que não surpreende devido a esse campo de estudo ser novo. (MUZIO et al 2001, p.2).

Os Objetos de Aprendizado são definidos por WILEY (2001) como sendo elementos computacionais de instrução baseados no paradigma de orientação a objetos da ciência da computação. A sua construção é feita através de componentes instrucionais que podem ser reutilizados inúmeras vezes em diferentes contextos de aprendizagem.

Já o IEEE vai mais além, segundo esta instituição os Objetos de Aprendizagem podem ser: "Qualquer entidade, digital ou não digital, que possa ser usada, reutilizada ou referenciada durante o uso de tecnologias que suportem ensino" (IEEE, 2010).

Os estudos sobre OA são recentes, de forma que não há um consenso universalmente aceito sobre sua definição. Os OA podem ser criados em qualquer mídia ou formato, podendo ser simples como uma animação ou uma apresentação de slides ou complexos como uma simulação. Os Objetos de Aprendizagem utiliza-se de imagens, animações e applets, documentos VRML (realidade virtual), arquivos de texto ou hipertexto, dentre outros. Não há um limite de tamanho para um Objeto de Aprendizagem, porém existe o consenso de que ele deve ter um propósito educacional definido, um elemento que estimule a reflexão do estudante e que sua aplicação não se restrinja a um único contexto (BETTIO & MARTINS, 2004).

Diversos fatores favorecem o uso de Objetos de Aprendizagem na área educacional, dentre eles pode-se citar segundo Longmire (2001):

**Flexibilidade:** os Objetos de Aprendizagem podem ser construídos para uso em múltiplos contexto, por isso, podem ser reutilizados de forma mais fácil.

**Customização:** Os conceitos no qual os AO são criados permitem que eles possam ser utilizados em diversas situações, com diversas metodologias de forma customizada, sendo assim adaptado para qualquer tipo de curso.

**Interoperabilidade:** os OA por ser desenvolvidos com base na orientação a objetos possuem padrões especificados em cada projeto, e esses padrões permitem que eles possam ser utilizados em qualquer sistema de aprendizagem.

**Aumento do valor do conteúdo:** a possibilidade de reutilização dos Objetos de Aprendizagem possibilita que estes agreguem diversos tipos de conhecimento aumentando assim o seu valor.

Após criação dos Objetos de Aprendizagem os autores os disponibilizam em algum repositório, para que os objetos desenvolvidos sejam utilizados de forma colaborativa por professores de qualquer parte do mundo. Dentre esses repositórios tem-se o Banco Internacional de Objetos Educacionais que foi criado pelo Ministério da Educação em parceria com o Ministério de Ciência e Tecnologia, Rede Latino America de Portais Educacionais – RELPE e outras instituições. Esse repositório tem por objetivo manter e

compartilhar diversos conteúdos pedagógicos digitais disponibilizados em várias línguas. Tais conteúdos primam por estimular o raciocínio e o pensamento crítico dos estudantes, associando o potencial da informática às novas abordagens pedagógicas.

Outro importante repositório de AO é o Projeto CESTA - Coletânea de Entidades de Suporte ao uso de Tecnologia na Aprendizagem. Desenvolvido na Universidade Federal do Rio Grande do Sul (UFRGS) com o propósito de armazenar os Objetos criados na instituição, principalmente os materiais que podem ser utilizados na modalidade de ensino à distância (CINTED, 2010). Já Labvirt, projeto desenvolvido na Universidade de São Paulo (USP), tem por objetivo agrupar objetos, desenvolvido para as áreas de Física e Biologia, no portal do projeto também são disponibilizados links de simulações, discussões, fórum e outras informações, além de possibilitar a inserção de novos objetos e simulações.

### **3. DIFICULDADES DO ENSINO-APRENDIZAGEM DE FUNDAMENTO DE PROGRAMAÇÃO**

Aprender a programar é um processo difícil e exigente. Para se tornarem bons programadores os alunos têm de adquirir um conjunto de conhecimentos que vão além de conhecer a sintaxe e a semântica de uma linguagem de programação (GOMES & MENDES, 2000). Eles necessitam de aprender a resolver problemas e elaborar algoritmos, o que dificulta a aprendizagem da programação. De fato, muitos deles sentem dificuldades nesse processo, o que resulta num elevado índice de insucesso às disciplinas de programação, que surgem geralmente no início dos cursos voltados a área da computação.

Nos últimos anos, essas disciplinas têm apresentado altos índices de desistência e reprovação, normalmente nas disciplinas com ênfase em Programação, Algoritmos e Estruturas de Dados. Essas reprovações geram desmotivação nos alunos, prejudicando o processo de ensino-aprendizagem. A evasão, além do afastamento dos alunos do seu objetivo principal que é a formação intelectual e profissional, gera desconfiança na comunidade acadêmica quanto à qualidade de tais cursos superiores, impedindo a entrada de novos alunos e retardando o crescimento da área de computação (RODRIGUES, 2004).

Muitos dos problemas associados a este processo é bastante questionável uma vez que envolvem ambas as classes de atores, tanto os alunos quanto os professores, as metodologias aplicadas e até certo grau de dificuldade intrínseco da área.

Alguns alunos não conseguem desenvolver o raciocínio lógico necessário para o posterior desenvolvimento de programas, sendo a abstração de ideias uma das principais dificuldades de programadores iniciantes. O nível de dificuldade no processo faz com que haja baixa motivação dos alunos, apatia, baixa auto-estima culminando com a evasão e reprovação (SCHULTZ 2003; CHAVES DE CASTRO et al 2003; DELGADO et al 2004).

A questão chave no processo é justamente como motivar o aluno, fazê-lo tomar gosto pelo aprendizado procurando superar suas dificuldades como a falta de habilidades matemáticas. O modo tradicional de apresentar o conteúdo de tal disciplina se faz de forma expositiva concentrando-se na solução de problemas. O professor apresenta o conteúdo através do chamado portugol (pseudo-código), desenvolve alguns exemplos e propõe exercícios a turma. O modo tradicional não consegue facilmente motivar os alunos a se interessarem pela disciplina, pois não é claro para os alunos a importância de certos conteúdos de sua formação (KOLIVER *et al*, 2004; NOBRE & MENEZES 2002).

O relacionamento entre o professor e o aluno é um dos fatos decisivos para a motivação do aluno, com essa harmonia o aluno pode demonstrar interesse em um ambiente descontraído e que colabora para o seu aprendizado, onde o professor preocupa-se em mostrar o que sabe. Além disso, uma metodologia pedagógica adequada contribui para o aprendizado,

porém, como afirma SANTOS *et al* (2003), poucas metodologias são universalmente aceitas e, mesmo as mais consolidadas não formalizam um processo claro de procedimento de ensino, deixando muitos aspectos delegados ao professor. Neste contexto deve-se questionar a diversidade de ritmos de aprendizagem dos alunos conjugada, muitas vezes, com turmas grandes (TOBAR *et al*, 2001). A partir de problemas apresentados anteriormente, ficou claro que o ensino da programação é feito de forma convencional como as outras áreas do conhecimento, de forma a apresentar os conteúdos, aplicar alguns exemplos básicos, trabalhar mais teorias, aumentar o nível dos problemas, e assim por diante; trabalhando os alunos da mesma forma e no mesmo grau de entendimento. Essa forma de trabalhar a programação desmotiva o aluno lavando ao afastamento do curso (BORGES, 2000).

De forma geral, as dificuldades de aprender e ensinar programação não são tão fáceis de solucionar, porém podem-se amenizar esses problemas com melhorias pedagógicas e maneiras interativas e intuitivas ao ensino da programação por parte dos professores e dos alunos. Uma dessas formas pode ser através da utilização de ferramentas computacionais que o auxiliem nesse processo de aprendizagem. Nesse contexto Botelho (2008) apresenta os Sistemas Tutores Inteligentes (STI) como principais ambiente telemáticos que auxiliam no aprendizado da programação. Os STI são sistemas computacionais que unem técnicas de IA com teorias pedagógicas para tutorar um aluno em determinada disciplina, como por exemplo a programação.

### 3.1. EXEMPLOS DE SISTEMAS TUTORES INTELIGENTES PARA O ENSINO DA PROGRAMAÇÃO

Diversos sistemas foram desenvolvidos para auxiliar alunos e professores no processo de aprendizagem da programação. Alguns desses sistemas são apresentados por Petry (2005), Botelho (2008) e Nobre e Menezes (2002), que serão descritos a seguir.

**Lisp Tutor:** STI desenvolvido na Carnegie-Mellon University, para acompanhar o aluno durante o processo de resolução dos problemas e transcrição do código. O sistema possui uma sequência de regras de produção que é repassada para o aluno à medida que ele vai resolvendo o problema, caso haja deficiências durante a produção o aluno é informado e não é permitida a continuação até seja resolvido aquele problema. Essa verificação do código é realizada quando o aluno aciona o tutor. A interação com o usuário é realizada através de um editor de textos e uma interface para o planejamento de programas que sejam mais complexos.

**SQL-Tutor:** Sistema Tutor criado para o ensino da linguagem de banco de dados SQL. O sistema faz uma avaliação dos erros cometidos pelo aluno nas linhas de código e demonstra a correta solução. O sistema possui uma interface com o usuário; um módulo pedagógico, que é responsável por selecionar o problema, verificar os erros e gera as mensagens de erros apropriadas; e um módulo do estudante que é responsável por registrar o uso das restrições.

**Algo-LC:** STI desenvolvido na UFSC, que se diferencia dos outros Sistemas Tutores pelo fato de ter um Companheiro de Aprendizagem, companheiro este que apóia o aluno durante a resolução dos exercícios, enviando mensagens o estimulando a verificar seus erros e corrigi-los e não demonstrando a resolução, como é feito em outros STI.

## 4. SUGESTÕES DE UTILIZAÇÃO DE OA EM STI PARA O ENSINO DA PROGRAMAÇÃO

Os problemas encontrados no processo de ensino/aprendizagem da programação ocorrem devido à diversos fatores, mas pode-se destacar a motivação com um ponto

determinante, pois os alunos são desmotivados a aprenderem o conteúdo através das didáticas utilizadas pelos professores. Logo, aplicar novas formas de aprendizado é essencial, por isso pode-se utilizar estratégias onde os alunos possam aprender de forma lúdica, brincando, sem ter a obrigação de aprender.

Após os estudos efetuados não foram detectados nos Sistemas Tutores Inteligentes analisados nenhuma característica dessa forma de aprendizagem lúdica, assim sugere-se que sejam utilizados objetos de aprendizagem nos STI para auxílio do ensino a programação, como forma de estimular os alunos a terem uma aprendizagem diferenciada. Os OA's podem ser utilizados nesses sistemas, pois são construídos com características da orientação a objetos e são independentes, possibilitando ao professor montar diferentes tipos de estratégias de aprendizado, adaptados para cada tipo de turma ou de alunos.

Dentro do projeto iniciado surgiu a idéia de utilizar cenários e enredos voltados para o contexto da região amazônica, já que há uma diversidade enorme de histórias, lendas, culturas e costumes nessa região. A escolha desse tema para os enredos foi devido à necessidade de apresentar os conteúdos da programação de forma clara e lúdica principalmente para os alunos dessa região. Um exemplo de um protótipo de Objeto de aprendizagem é apresentado na Figura 1 que apresenta a criação de um objeto no software Blender<sup>1</sup>. Este está sendo desenvolvido em forma de um Paneiro, que é um cesto confeccionado com palha de palmeiras e é utilizado para armazenagem e transporte de Açaí e Castanha do Pará.



**Figura 1:** Protótipo de OA Paneiro em desenvolvimento no Blender

O objeto desenvolvido será utilizado em diversos cenários, um deles é o que o aluno deverá montar algoritmos que permitam um personagem colher o açaí nas suas altas palmeiras. E para a criação desses algoritmos ele terá a disposição dentro do objeto Paneiro estruturas que representam conceitos da lógica da programação, como é demonstrado na Figura 2.

---

<sup>1</sup> Blender : é um programa de computador de código aberto, desenvolvido pela Blender Foundation, para modelagem, animação, texturização, composição, renderização, edição de vídeo e criação de aplicações interativas em 3D, tais como jogos.



- BORGES, M.** Avaliação de uma metodologia alternativa para a aprendizagem de programação. VIII Workshop de Educação em Computação – WEI 2000. Curitiba, 2000
- BOTELHO, Carlos Alberto.** Sistemas Tutores no domínio da programação. Revista Informática Aplicada v IV nº1 jan/jun 2008.
- CINTED.** CESTA - Coletânea de Entidades de Suporte ao uso de Tecnologia na Aprendizagem. Disponível em: < <http://www.cinted.ufrgs.br/CESTA/cestadescr.html>>. Acesso em 08 jul 2010.
- CHAVES DE CASTRO, T., CASTRO JÚNIOR, A., MENEZES, C., BOERES, M. e RAUBER, M.** “Utilizando Programação Funcional em Disciplinas Introdutórias de Computação”. XI Workshop de Educação em Computação – WEI 2003. Campinas, 2003.
- DELGADO, C., XEXEO, J. A. M., SOUZA, I. F., CAMPOS, M., RAPKIEWICZ, C. E.** “Uma Abordagem Pedagógica para a Iniciação ao Estudo de Algoritmos”. XII Workshop de Educação em Computação (WEI2004). Salvador, 2004. Revista Informática Aplicada
- JESUS, Alexandre Negrão de; LOPES, Daniel Lourenço, PERIN, Fernanda Rezende; CANTÃO, Juliana Martin; PINHEIRO, Edson.** Objetos de Aprendizagem no ensino da Lógica de Programação. Revista Informática Aplicada, v. III nº 2 jul/dez 2007.
- GOMES, A. J. e MENDES, A. J.** Suporte à aprendizagem da programação com o ambiente SICAS, (2000)In: V Congresso Ibero-Americano de Informática Educativa, Viña del Mar, 2000.
- KOLIVER, C.; DORNELES, R. V.; CASA, M. E.** “Das (muitas) dúvidas e (poucas) certezas do ensino de algoritmos”. XII Workshop de Educação em Computação (WEI2004). Salvador, 2004.
- IEEE Learning Technology Standards Committee (LTSC).** Learning Objects Metadata (LOM) (2000). Disponível em:<<http://www.ieeeltsc.org>>. Acesso em 15 jul 2010.
- LONGMIRE, W.** A Primer On Learning Objects. American Society for Training & Development. Virginia. USA. 2001.
- MUZIO, J.; HEINS, T.; MUNDELL, R.** Experiences with Reusable e-Learning Objects: From Theory to Practice. Victoria, Canadá. 2001.
- NOBRE, I. A. M. N., MENEZES, C. S.** “Suporte à Cooperação em um Ambiente de Aprendizagem para Programação (SAmbA)”. XIII Simpósio Brasileiro de Informática na Educação – SBIE 2002. São Leopoldo, 2002.
- PETRY, Patrícia Gerent.** Um sistema para o ensino e aprendizagem de algoritmos utilizando um companheiro de aprendizagem colaborativo. 2008. 86p. Tese (Mestrado em Ciência da Computação), Universidade Federal de Santa Catarina. Florianópolis, 2005.
- RODRIGUES, M.** Experiências positivas para o ensino de algoritmos. IV Escola Regional de Computação Bahia-Sergipe. Feira de Santana, 2004.
- SANTOS, G.; DIRENE, A. I.; GUEDES, A. L. P.** “Autoria e Interpretação Tutorial de Soluções Alternativas para Promover o Ensino de Programação de Computadores”. XIV Simpósio Brasileiro de Informática na Educação – SBIE 2003. Rio de Janeiro, 2003.
- SCHULTZ, M. R. O.** Metodologias para Ensino de Lógica de Programação de Computadores. 2003. 69p. Monografia (Especialização em Ciência da Computação). Universidade Federal de Santa Catarina, Florianópolis, (2003).
- TOBAR, C. M., ROSA, J. L. G., COELHO, J. M. A., PANNAIN, R.** “Uma Arquitetura de Ambiente Colaborativo para o Aprendizado de Programação”. XII Simpósio Brasileiro de Informática na Educação (SBIE 2001). Vitória, 2001.
- Wiley, D.** Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. The Instructional Use of Learning Objects. Wiley, D. (ed.) (2001). Disponível em:<<http://www.reusability.org/read/chapters/wiley.doc>>. Acesso em: 01 jul 2010.