

TBC-AED e TBC-AED/WEB: Um Desafio no Ensino de Algoritmos, Estruturas de Dados e Programação^{*}

Rodrigo Pereira dos Santos¹, Heitor Augustus Xavier Costa²

Departamento de Ciência da Computação – Universidade Federal de Lavras (UFLA)
Caixa Postal 3037 – 37.200-000 – Lavras – MG – Brasil

¹rpsantos@comp.ufla.br, ²heitor@ufla.br

Abstract. *The paper objective is to present TBC-AED and its evolution, TBC-AED/WEB, an educational information system that contains a lot of themes related to the taught contents to students in Computation, now available in applets-Java for Web. The paper contributes to literature and to teachers concerning the importance of a good teaching of algorithms and programming and of the use of software in the education. Therefore, teaching strategies and computing tools are united, trying to help the improvement of teaching quality of programming, algorithms and data structures in the graduation courses in Computing.*

Key-words: *algorithms and data structures, programming, computing in education.*

Resumo. *Este artigo tem o objetivo de apresentar o TBC-AED e sua evolução, o TBC-AED/WEB, que consistem em um sistema de informação educacional que contém vários temas relacionados aos conteúdos lecionados aos iniciantes em Computação, agora disponíveis em applets-Java para Web. Procura também deixar uma contribuição à literatura e aos docentes da área acerca da importância de um bom ensino de algoritmos e programação e do uso de produtos de software na educação. Portanto, estratégias de ensino e ferramentas computacionais são unidas, buscando contribuir para a melhoria da qualidade de ensino de programação, algoritmos e estruturas de dados nos cursos de graduação em Computação e Informática.*

Palavras-chaves: *algoritmos e estruturas de dados, programação, informática na educação.*

1. Introdução

A crescente quantidade de informação e a tecnologia gerada tornam necessária a presença de profissionais de caráter persuasivo e inovador, que sejam capazes de unir inteligência e dinamismo na busca de melhores soluções para o mercado. Isso encontra sua principal base nas metodologias de ensino aplicadas durante o estágio universitário, principalmente na área de Computação e Informática, que envolve tecnologias em constante desenvolvimento.

Com o intuito de produzir melhores resultados no processo de aprendizagem nessas áreas, faz-se constante a necessidade do uso de didáticas de ensino de forma geral. O objetivo é procurar transformar processos abstratos em concretos, através de produtos de *software* que possibilitem melhor compreensão dos tópicos abordados e maior

^{*} Apoio PBICT/FAPEMIG.

interação entre aluno-objeto de trabalho (neste caso, o computador). Isso encontra sua maior barreira nos estágios iniciais dos cursos relacionados, quando do contato com **o estudo de algoritmos, o raciocínio lógico e a programação**.

Atualmente, o ensino de algoritmos busca nas Ciências Exatas seu pilar de sustentação, pois disciplinas nessa área despertam o raciocínio matemático-lógico para resolução de problemas. A título de exemplo, conforme [Júnior e Rapkiewicz 2004], alguns autores preconizam a necessidade de domínio de habilidades matemáticas prévias ou ao menos integradas, ou pelo menos desejável, sendo considerado até um bom indicador de sucesso para o processo. Para outros, o próprio nível de conhecimento prévio de lógica matemática é discutido. Mas, isso deve ser mais uma das ferramentas para o estudante e não a sua única fonte, pois a sua formação futura será muito ampla.

O conceito de programação, segundo [Wirth 1989], é a arte ou a técnica de construir e formular algoritmos de uma maneira sistemática. Esse conceito induz que o ingresso tem um dom de programar ou ele pode aprender as técnicas necessárias a aprimorar seu conhecimento. Logo, o iniciante deve ter sua base bem sedimentada, para que fique apto a prosseguir de maneira positiva durante os seus estudos. Com isso, professores de disciplinas relacionadas à programação sentem a grande responsabilidade de estar aperfeiçoando sua maneira de conduzir a estrutura disciplinar dos graduandos em seus estágios iniciais da universidade.

Uma linha de pesquisa, visando melhorias no ensino básico em Computação e Informática é direcionada para a tendência de utilizar ferramentas computacionais como ambientes de estudo. Segundo [Soares *et al.* 2004], a partir de observações dentro de disciplinas de graduação, percebe-se também um melhor resultado no aprendizado por meio de atividades práticas de desenvolvimento de simuladores e ferramentas visuais didáticas de representação de conceitos abstratos. Além disso, a linha acopla contribuições dos pesquisadores a novas estratégias de transmissão de conhecimentos. Isso promove a ampliação do acervo existente, a melhor organização de currículos para o mercado de trabalho e o acesso facilitado destes recursos através de suporte via *Web*.

Em um estudo anterior, foi desenvolvido o *TBC-AED* (Treinamento Baseado em Computador para Algoritmos e Estruturas de Dados), que foi utilizado pela primeira vez na turma de 2º período do Curso de Ciência da Computação da Universidade Federal de Lavras (1º semestre de 2005). Ele é um ambiente computacional didático, fruto de um estudo sobre algoritmos, estruturas de dados e programação. O *TBC-AED* visa tornar o ensino deste conteúdo mais prático e abrangente, de forma a despertar o interesse do aluno e o seu espírito de pesquisa. Assim, seu rendimento ao longo do curso aumentará, melhorando o seu desempenho em disciplinas mais específicas. Pôde-se notar que o *TBC-AED* despertou a atenção dos alunos de forma inesperada, devido à inovação e à dinamização, que foram injetadas às aulas.

O objetivo deste artigo é apresentar o *TBC-AED* e sua evolução, o *TBC-AED/WEB*, além de relatar a importância de construir uma boa base de algoritmos e programação para uma melhor conduta em períodos avançados de cursos de Computação. Também retrata algumas implicações acerca do uso de informática em educação. Dessa forma, pretende-se possibilitar maior disseminação do conhecimento e promover a interatividade entre docentes e graduandos da área, enriquecendo a pesquisa com críticas e sugestões.

O artigo está organizado da seguinte forma: a seção 2 apresenta a importância da programação e dos algoritmos no ensino de Computação; a seção 3 conceitua Informática na Educação e analisa os impactos do uso de produtos de *software* no ensino; a seção 4 retrata a importância do uso de ferramentas computacionais no ensino e detalha algumas ferramentas disponíveis atualmente para o ensino de algoritmos, estruturas de dados e programação; a seção 5 apresenta o *TBC-AED* e o *TBC-AED/WEB*; e a seção 6 inclui resultados iniciais de sua aplicação. Por fim, são apresentadas as conclusões e algumas sugestões de trabalhos futuros.

2. Programação e Algoritmos: a Base da Computação

Segundo a estrutura curricular presente nas Diretrizes Curriculares do MEC [Azeredo 2000], a matéria de Programação faz parte da área de formação básica em Ciência da Computação. Seu conteúdo abrange, além do ensino de linguagens de programação, os conceitos, os princípios e os modelos de programação e o estudo de estruturas de dados e de métodos de classificação e pesquisa de dados. Conforme consta na caracterização da matéria programação de computadores, é uma atividade voltada à solução de problemas. Ela está relacionada com uma variada gama de outras atividades como especificação, projeto, validação, modelagem e estruturação de programas e dados, utilizando-se das linguagens de programação como ferramentas.

Ao contrário do que se apregoava há alguns anos, a atividade de programação deixou de ser uma arte para se tornar uma ciência, envolvendo um conjunto de princípios, técnicas e formalismos que visam à produção de produtos de *software* bem estruturados e confiáveis. Cite-se, dentre estes, os princípios da abstração e do encapsulamento e as técnicas de modularização e de programação estruturada.

Portanto, o estudo de programação não se restringe ao estudo de linguagens de programação. As linguagens de programação constituem-se em uma ferramenta de concretização de produtos de *software*, que representa o resultado da aplicação de uma série de conhecimentos que transformam a especificação da solução de um problema em um programa de computador que efetivamente resolve aquele problema. Deve ser dada ênfase aos aspectos funcionais e estruturais das linguagens, em detrimento aos detalhes de sintaxe. O estudo de linguagens deve ser precedido do estudo dos principais paradigmas de programação, notadamente a programação imperativa, a funcional, a baseada em lógica e a orientada a objetos.

O desenvolvimento de algoritmos e o estudo de estruturas de dados devem receber especial atenção na abordagem do tema programação. Igualmente, deve ser dada ênfase ao estudo das técnicas de especificação, projeto e validação de programas. Deve-se entender que ensinar programação não é apenas ensinar uma linguagem de programação. Este ensino envolve entender problemas e descrever formas de resolução, de maneira imparcial, para que então sejam codificadas em uma linguagem. Ou seja, somente após o aprendizado dos conceitos de algoritmo e fundamentos de lógica, o estudante pode travar contato com uma linguagem de programação. Além disso, o estudo de programação deve passar efetivamente pelo estudo de estruturas de dados. Ou seja, quando se projeta um programa, deve-se pensar tanto nos algoritmos que manipulam os dados como também na forma de estruturar estes dados. Desta maneira, as duas coisas se complementam.

De acordo com [Azeredo 2000], dentro do ensino de estruturas de dados, deve-se dar destaque à parte conceitual e comportamental das estruturas, antes de pensar em implementação. Por exemplo, no ensino de estruturas lineares, deve-se transmitir ao aluno o seu conceito, através das operações aplicáveis, como criar, acrescentar, consultar e remover elementos. Finalmente, discutem-se formas de implementações ou representações físicas das listas. Deve ser dada ênfase especial no estudo de estruturas encadeadas. Nas estruturas encadeadas, o aluno apresenta maior dificuldade de assimilação do conceito, uma vez que o encadeamento não é um conceito muito freqüente na vida real, dificultando sua compreensão. Dessa forma, é recomendável que as disciplinas de estruturas de dados possuam significativa atividade prática em laboratório.

Quanto ao sistema de avaliação aplicado, é o mesmo adotado tradicionalmente na maioria dos cursos de natureza teórico-prática. Além de provas periódicas, são realizados alguns trabalhos de programação para fixação. Uma sugestão do uso de um produto de *software* durante as aulas é a redução do peso em provas teóricas, normalmente elaboradas sobre um conteúdo denso.

Com relação aos cursos na área de Computação e Informática, os cursos de licenciatura visam a formação de educadores para o ensino médio. Os cursos de Bacharelado em Ciência da Computação e em Engenharia de Computação são voltados para a formação de recursos humanos para o desenvolvimento científico e tecnológico da computação. Não necessariamente devem sair dos cursos com o perfil de programadores. Os cursos de Sistemas de Informação incluem a formação de programadores entre seus objetivos. Em um curso de Licenciatura em Computação, o aluno poderá ter uma carga de prática de programação menor, sem prejuízo da parte conceitual da matéria. Deve haver uma diferenciação entre os casos de cursos de atividade fim e atividade meio. Para este último, recomenda-se uma ênfase maior no conhecimento de linguagens de programação e atividades práticas.

Assim, a programação é, pois, um dos pontos chaves em um curso de Computação, pois é a atividade que supre o computador com meios para servir ao usuário. Seu uso adequado e o entendimento de conceitos, princípios, teorias e tecnologias podem conduzir à produção de produtos de *software* de qualidade, objetivo final, tanto quando se tem a computação como um meio como quando a computação é tida como um fim. Com esse intuito, [Setúbal 2000] apresenta algumas recomendações genéricas a serem aplicadas no ensino de disciplinas relacionadas a algoritmos:

- **coerência com os objetivos fundamentais.** Por coerência, entende-se que o professor deve: i) expressar claramente as idéias, os conceitos e as técnicas perante os alunos (se o professor coloca algoritmos confusos na lousa ou em transparências, ele não pode esperar algoritmos claros nas respostas dos alunos); ii) destacar a importância dos resultados teóricos e mostrar rigor formal toda vez que isto se fizer necessário; e iii) valorizar o uso de técnicas na resolução de problemas;
- **ênfase no pensamento crítico.** Deve-se ter um cuidado especial, pois os alunos que têm pouca maturidade matemática tendem a acreditar em qualquer demonstração. Tal comportamento deve ser desestimulado. É essencial que os alunos duvidem daquilo que é apresentado a eles e é com dúvidas saudáveis e sua resolução que a percepção da importância do resultado teórico poderá ser consolidada. Nesse sentido, considera-se um recurso valioso o conjunto de exercícios que pedem para os alunos

identificarem falhas de argumentação, erros em algoritmos ou erros em notícias da imprensa;

- **a teoria na prática.** A experiência mostra que os alunos em geral não se sentem atraídos, por considerarem-na muito abstrata. Por esse motivo, crê-se ser importante usar como recurso didático exemplos reais. A inclusão de projetos de implementação, em disciplinas teóricas ou em disciplinas específicas, torna a matéria menos abstrata. De resto, é importante salientar para os alunos o grande impacto que os resultados teóricos têm alcançado na prática.

3. Informática na Educação e *Software* Educacional

Informática na Educação é uma das áreas de Computação e Informática, a qual trata da utilização de computadores para o ensino, como forma de ampliar a difusão de conhecimentos e facilitar a assimilação de informações.

Entretanto, conforme [Zambalde e Alves 2002], a incorporação da informática nos ambientes educacionais provoca mudanças. Nesse sentido, deve-se sempre questioná-las e nem sempre adotá-las. O objetivo da introdução da informática no ambiente educacional não deve ser o modismo ou a atualização com relação às inovações tecnológicas e ao mercado. O questionamento é imprescindível, visto que as alterações podem alcançar estrutura física, equipamentos, cultura, forma de aprendizado, modos de comunicação, enfim, processos dos mais variados tipos.

Para [Almeida e Nogueira 2001], a utilização de computadores na escola pode contribuir muito para um ensino de melhor qualidade, além de facilitar a aprendizagem. Para isso, é necessário que os professores sintam-se estimulados e preparados para enfrentar esta nova realidade. É preciso oferecer subsídios, recursos e capacitação. Com isto, poderão sentir-se capazes de criar seus próprios produtos de *software*, suas próprias *home-pages*, conforme sua metodologia e disciplina, visto que é difícil encontrar alunos que não utilizam o computador para realizar tarefas escolares, pesquisa e/ou lazer. Cabe à escola preparar seus docentes para enfrentar esta nova realidade.

De fato, sabe-se que a informática está sendo inserida na educação pela necessidade de transpor barreiras do educar convencional, pois tudo se modernizou na educação. Até o advento da informática se tornou convencional, frente a esta nova forma pedagógica de educação, oportunizando às escolas uma renovação de trabalhar os conteúdos programáticos, propiciando ao aluno eficiência na construção do conhecimento, convertendo a aula em um espaço real de interação, de troca de resultados, adaptando os dados à realidade do educando ([Koslowski e Nunes 2001]).

Entretanto, em [Koslowski e Nunes 2001], o uso da informática na educação não significa a soma da informática e da educação, mas a integração destas duas áreas. Para haver integração, é necessário o domínio dos assuntos que estão sendo integrados. Seguindo esta linha de pensamento, pensar em computadores na educação não significa pensar na máquina, mas sim na educação. Educação e Informática devem ser consideradas como um todo, visando o benefício da sociedade. Apesar dos aspectos positivos da utilização da informática na escola, percebe-se que a maioria das escolas ainda está a passos lentos e tem dúvidas sobre a real contribuição dos computadores ao ensino.

No entanto, a mesma forma aditiva pela qual tem sido pensada a introdução de computadores na educação também vem se aplicando ao processo de preparação de

professores. Tal preparação realiza-se através de cursos ou treinamentos de pequena duração, para exploração de determinados produtos de *software*. Resta ao professor desenvolver atividades com essa nova ferramenta junto aos alunos, mesmo sem ter a oportunidade de analisar as dificuldades e as potencialidades de seu uso na prática pedagógica. Os alunos, por crescerem em uma sociedade permeada de recursos tecnológicos, são hábeis manipuladores da tecnologia e a dominam com maior rapidez e desenvoltura que seus professores. Mesmo os alunos pertencentes a camadas menos favorecidas têm contato com recursos tecnológicos na rua e na televisão e a sua percepção sobre tais recursos é diferente da percepção de uma pessoa que cresceu em uma época em que o convívio com a tecnologia era muito restrito ([Almeida 2000]). Portanto, deve-se levar em consideração ainda que os professores treinados apenas para o uso de certos recursos computacionais são rapidamente ultrapassados por seus alunos, que têm condições de explorar o computador de forma mais criativa, e isso provoca diversas indagações quanto ao papel do professor e da educação.

Nesse contexto, surge o instrumento computacional a ser utilizado para educar pessoas. Tal recurso é o produto de *software* educacional. Consoante [Zambalde e Alves 2002], o produto de *software* educacional é um programa desenvolvido para atender aos objetivos educacionais previamente estabelecidos e, para que ele seja efetivo e esteja à altura das necessidades pedagógicas, é necessário que seu desenvolvimento conte com especialistas das áreas de Educação e Informática. Mas os produtos de *software* educacionais possuem pontos fortes e limitações. É preciso saber quando ele é adequado para os objetivos curriculares e pode, por isso, ser integrado ao contexto educacional. O professor deve adotar o produto de *software* que instigue as habilidades cognitivas dos alunos e, acima de tudo, lhes ofereça situações para que possam transferir seus conhecimentos para a solução de novos problemas ([Zambalde e Alves 2002]).

Entende-se que várias são as ferramentas de autoria encontradas para facilitar a criação de aplicações educacionais nas mais diversas áreas do ensino, mas em geral todas buscam a criação de produtos de *software* multimídia. O uso dessas ferramentas aplicadas à educação teve um elevado crescimento nos últimos anos, mas, apesar da riqueza visual e sonora que comporta, faz com que o aprendiz interaja com o ambiente como se estivesse manuseando um livro eletrônico. Na maioria dos casos, é pouca ou quase nula a possibilidade do aprendiz intervir nesse ambiente para criar ou interagir de forma ativa sobre o objeto de conhecimento. Diante disso, buscou-se um paradigma alternativo, onde o aprendiz interage ativamente na construção de conhecimento, em ambientes tais como o de jogos, nos quais os participantes são personagens de uma história que se desenrola e suas ações vão modificar esta história à medida que ela acontece ([Souza e Wazlawick 1997]).

Dessa forma, existem diferentes formas de utilizar o computador na educação e, infelizmente, é mais comum encontrar a sua utilização como objeto de estudo, como facilitador na realização de determinadas tarefas ou como máquina de ensinar que repetem os métodos tradicionais de ensino, conhecidos como instrucionistas. Está-se interessado no uso do computador como objeto que contribua no processo de aprendizagem. Isso ocorre por vários motivos. Em primeiro lugar, existe uma exigência muito grande da sociedade para que os indivíduos saibam manipular o computador, pois ele está presente em quase todas as nossas atividades diárias e, portanto, uma pessoa que não consiga conviver com esse objeto ficará inevitavelmente à margem de um mundo cada vez mais dominado por recursos tecnológicos. Esse argumento tem a sua verdade,

mas por outro lado, os educadores não podem se preocupar apenas com esse lado da questão, bastante limitada, voltada apenas para o treinamento de indivíduos na utilização de produtos de *software*. Em segundo lugar, cresce cada vez mais o interesse em conseguir novos mercados de consumidores de produtos de *software* educativos e esses aparecem com os mais variados atrativos gráficos, porém pobres do ponto de vista pedagógico. Diante da falta de conhecimento por parte da população, essa indústria tem encontrado um campo bastante receptivo a esse tipo de produto.

Ainda assim, não se deve considerar como digno de uso no ambiente escolar apenas os produtos de *software* construídos especificamente para fins pedagógicos. Além disso, às vezes, eles não possuem uma proposta pedagógica clara. Dentro da visão que se tem de informática educativa, onde o computador deve ser usado com ferramenta cognitiva, entende-se que deve selecionar aplicativos que viabilizem esta forma de uso.

Mas, para usufruir todo o aparato tecnológico disponível, deve-se considerar que uma inovação imposta ou planejada por organismos externos à instituição é incompatível com a concepção dialética de inovação, pois tende a produzir rejeição ou a adicionar quantitativamente o novo instrumento ao arsenal disponível. As práticas impostas visam à otimização do ensino e não deixam espaço para o desenvolvimento de processos criativos. Isso foi observado com a introdução dos recursos audiovisuais nas escolas e, hoje, muitas instituições adotam esse procedimento em relação aos microcomputadores ([Almeida 2000]). Isso leva à necessidade de preparação de pessoal que saiba lidar com a iniciação à tecnologia, sobretudo professores do 2º e 3º graus. Logo, a preocupação com a formação de professores é uma constante nas universidades que desenvolvem trabalhos relacionados ao uso do computador em educação. O desenvolvimento de atividades de formação se faz por caminhos diferentes, que variam desde a criação de disciplinas específicas que tentam integrar informática e educação até a realização de cursos de pós-graduação.

Além disso, deve considerar que a formação não se encerra com a conclusão de cursos, oficinas ou outros eventos, pois deve ter o caráter de continuidade, que se concretiza por meio de reuniões periódicas, seminários, encontros presenciais e oficinas. Quaisquer que sejam as modalidades de formação escolhidas, sua concretização deve ser coerente com as necessidades do grupo em formação e prever espaço para o estabelecimento de conexões entre teoria, prática e domínio de recursos computacionais. Isso promove uma reorganização e uma transformação da prática pedagógica, segundo a reflexão favorecida pelo ciclo descrição-execução-reflexão-depuração.

4. A Importância do Uso de Ferramentas Computacionais no Ensino

Nesta seção, são apresentadas algumas ferramentas computacionais presentes na literatura e suas contribuições ao ensino de algoritmos, estruturas de dados e programação. Contudo, é importante ter em mente alguns pontos especiais:

- a aula de uma disciplina que relaciona programação, algoritmos e estruturas de dados deve ser realizada em laboratórios para que os alunos possam entender as abstrações apresentadas. Deve ser composta de uma parte teórica, onde conceitos são transmitidos, e uma parte prática, onde são entendidos comandos e tópicos ministrados. Deve também haver questões-desafio, para que os discentes sejam acostumados a usar de agilidade e criatividade na solução de problemas;
- o uso de produtos de *software* educacionais, que enfatizam animação gráfica, é de extrema importância como facilitador do processo de aprendizagem, uma vez que a

apresentação de conceitos abstratos se torna mais viável e didática, melhorando a qualidade do material das aulas;

- normalmente, um aluno se interessa por aulas diferenciadas e isso não só prende sua atenção, como também influenciam positivamente nas avaliações;
- uma economia de tempo pode ser conseguida e esta seria direcionada para maiores explicações e resoluções de exercícios, uma vez que o material didático seria virtual, evitando uma explanação cansativa através do quadro-negro;
- o aluno teria acesso livre ao material virtual para que pudesse estudar em casa. Além disso, ele possuirá um valioso projeto, para consultas futuras, em caso de necessidade – a tendência em abstrair detalhes, com o passar do tempo, sempre leva a dúvidas e estas a novas pesquisas em estruturas básicas ora aprendidas. Além disso, a existência de material via *Web* para estudos e pesquisas estimula maior contribuição tecnológica para a sociedade e maior alcance ao público. A divulgação das propostas de melhorias de processos educacionais visa injetar e obter informações para ampliação da literatura disponível;
- a transição de educadores seria facilitada, uma vez que existiria uma base pronta para ser apreciada, mantendo a qualidade de ensino e aprendizagem do conteúdo.

A seguir, são apresentados alguns projetos relacionados ao uso do computador para o ensino de algoritmos, estruturas de dados e programação. São eles:

- **ASTRAL** (*Animation of Data Structures and Algorithms*). É um ambiente de programação para produção de animações de algoritmos e de estruturas de dados com propósito instrucional. O ambiente foi desenvolvido no Instituto de Computação da UNICAMP. O uso do ASTRAL em disciplinas de graduação sobre estruturas de dados com componente de programação foi muito bem sucedido durante vários semestres ([Garcia *et al.* 1997]);
- **EDDL** (Estruturas de Dados Dinâmicas Lineares). Surgiu de uma discussão sobre estratégias de ensino das estruturas de dados e tipos de dados abstratos utilizados em programação. A sua estruturação e a concepção da sua funcionalidade englobam: i) uma abordagem introdutória aos conceitos ou especificação dos tipos de estruturas; ii) as técnicas de implementação; e iii) a utilização dessas estruturas na resolução de problemas. Oferece um conjunto de ferramentas de programação que o revelam ser ao mesmo tempo intuitivo e de fácil manipulação, diversificado, versátil e eficaz para o desenvolvimento de projetos, principalmente quando o aspecto gráfico é uma das componentes importantes a ter em conta ([Azul e Mendes 1998]);
- **TED** (Tutorial de Estruturas de Dados). É um tutorial para dinamizar o ensino na disciplina de estrutura de dados. Tem o objetivo de permitir uma maior interação entre o professor e o aluno, servindo como uma ferramenta de apoio ao aprendizado. Ele subdivide-se em várias estruturas de dados, permitindo ao aluno ter uma visão individualizada dos conteúdos, sem perder a visão do todo. Através de conceitos de dinamização de aprendizagem, como a visualização dos acontecimentos simultaneamente com o que é executado, o aluno cria uma visão de totalidade do sistema. O tutorial apresenta os algoritmos das estruturas de dados, ao mesmo tempo em que demonstra a sua implementação lógica através de exemplos visuais permitindo ao aluno acompanhar todos os passos simultaneamente ao que é executado ([Flávio 2004]);
- **Dictionary of Algorithms and Data Structures**. É um site que contém parte do *Software Quality Group do Software Diagnostics and Conformance Testing Division, Information Technology Laboratory*, que tem um dicionário de algoritmos, técnicas

algorítmicas, estruturas de dados, problemas atípicos e definições relativas. Alguns tópicos têm *links* para implementações ([Black 2005]).

5. TBC-AED e TBC-AED/WEB

Assimilar o conceito e o desenvolvimento de algoritmos é alvo de muitas dificuldades enfrentadas pelos alunos de cursos da área de Computação. Alunos que apresentam deficiências advindas dos ensinamentos fundamental e médio encontram grandes dificuldades em aprender conceitos simples de programação de computadores [Alves e Costa 2004].

Visando contribuir para a melhora deste quadro, em 2004 foi desenvolvido o *TBC-AED*, um produto de *software* que vem ao encontro dos objetivos discutidos até o presente momento. Ele analisa tópicos básicos de programação, englobando um conteúdo teórico sintético e direto, acompanhado de processo gráfico passo a passo. Isso facilita a visualização e o entendimento das informações apresentadas e disponibiliza maior tempo para a resolução de exercícios de aplicação e fixação. Além disso, pode-se alcançar maior interação entre o professor e os seus alunos, no sentido de aumentar o espaço para questionamentos.

Foi observado que o *TBC-AED*, mais que uma simples ferramenta, é um verdadeiro material de coleta de informações a respeito de programação, algoritmos e estruturas de dados. Ele representa uma experiência profissional, pois foi desenvolvido para fins de consumo, é gratuito e direcionado para alunos de área de Computação e Informática. Além disso, ele visa atender aos alunos que estudam algoritmos e estruturas de dados básicas. Isso é importante para tornar maior a interação graduando-computador, o que propicia estímulos de curiosidade, atenção e eficácia no desenvolvimento de trabalhos relacionados e do raciocínio lógico.

Durante o processo de embasamento teórico-prático, vários exemplos foram vistos e estudados para que o *TBC-AED* pudesse ser elaborado com cuidado especial, o que, *a priori*, não despertava qualquer tipo de receio; além disso, o seu conteúdo era de conhecimento. Com a introdução e a sedimentação de informações, o *TBC-AED* começou a ser desenvolvido e passou por várias etapas de melhorias, aplicadas tanto à forma de apresentação, quanto ao processamento dos algoritmos propriamente dito.

Enfim, o *TBC-AED* possui, diferentemente dos demais ambientes de ensino discutidos na seção anterior, algumas características, tais como: i) *links* explicativos, evitando a necessidade de aprendizagem via tutorial; ii) a usabilidade da interface gráfica é razoável, possibilitando ao professor apresentar conceitos iniciais, conteúdos teóricos e práticos aos poucos, como apresentaria em transparências; iii) conteúdo teórico simples, de forma a familiarizar melhor o aluno com o assunto; iv) processo gráfico passo a passo, com elementos numéricos, o que melhora a visualização e o entendimento; e v) legendas explicativas, que ilustram as etapas do processo de apresentação de algoritmos. Estas características facilitam o aprendizado e englobam aulas teóricas e práticas, ao envolver uma nova forma de transmitir conhecimento.

Quanto ao desenvolvimento, o *TBC-AED* apresenta os seguintes temas: Busca em Vetor (Busca Binária), Métodos de Ordenação (*Select Sort*, *Insert Sort*, *Bubble Sort*, *Merge Sort* e *Quick Sort*), Estruturas de Alocação Estática e Dinâmica de Memória (Lista, Fila e Pilha) e Árvores (Árvore Binária de Busca). Ele foi elaborado usando a linguagem de programação Java (devido principalmente à portabilidade) e o ambiente de desenvolvimento *NetBeans*. A Figura 1 apresenta a tela principal do *TBC-AED*.

Ao selecionar um desses temas, um *submenu* é apresentado, contendo um item explicativo do tema (Figura 2) e outros itens de representação gráfica do tema. A Figura 3 mostra uma janela relativa à representação gráfica do tópico *Busca Binária*, o qual apresenta uma breve introdução a respeito do assunto pesquisado (na parte superior), o algoritmo em *Portugol* (à esquerda), o painel de animação, onde o algoritmo será executado graficamente (à direita), contendo uma legenda e um conjunto de botões (na parte inferior). Exceto este, os demais tópicos contêm uma janela de mensagem interna usada para o efeito de passo a passo na parte inferior à direita (Figura 4).



Figura 1. Tela principal do TBC-AED

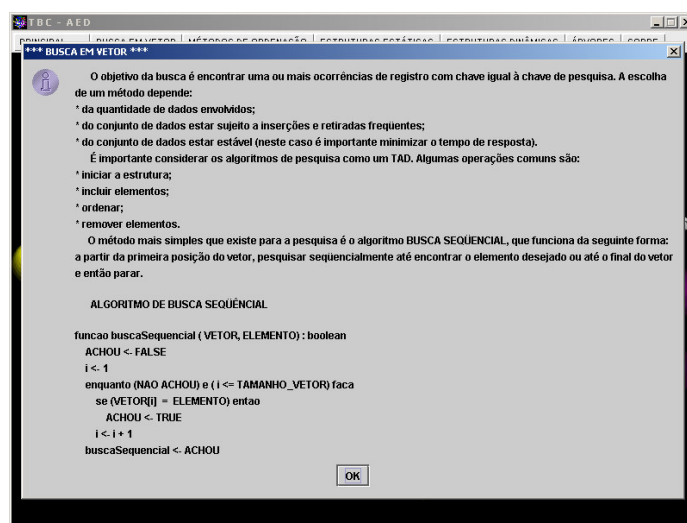


Figura 2. Janela de mensagem contendo uma introdução do tópico relacionado

A única tela que se diferencia dos padrões descritos anteriormente é relativa ao tópico *Árvore Binária de Busca* que, devido à sua peculiaridade de representação perante aos demais temas, demandou maior espaço para o processo gráfico (Figura 5). As mudanças ocorreram no posicionamento interno das informações: o algoritmo em *Portugol* possui uma área menor contendo na parte inferior uma janela de mensagem interna usada para o efeito de passo a passo, e o painel de animação é exibido em uma área maior, contendo na parte inferior um espaço para a exibição dos elementos quando o usuário requisitar uma das formas de imprimir uma árvore binária (pré-ordem, in-ordem ou pós-ordem) e uma legenda.

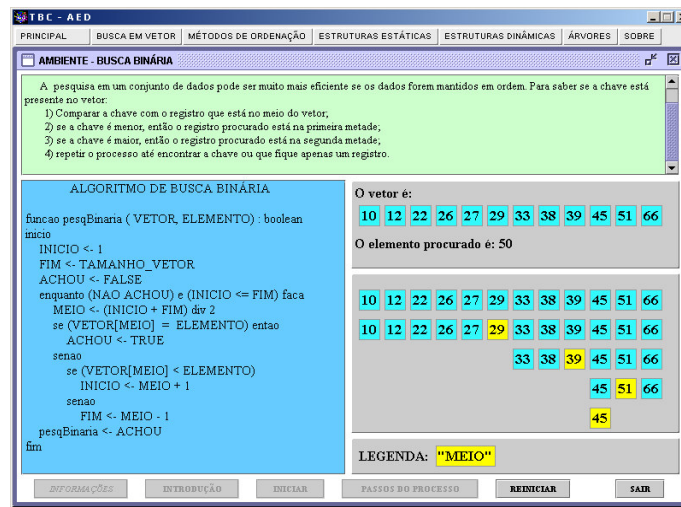


Figura 3. Tela *Busca Binária*

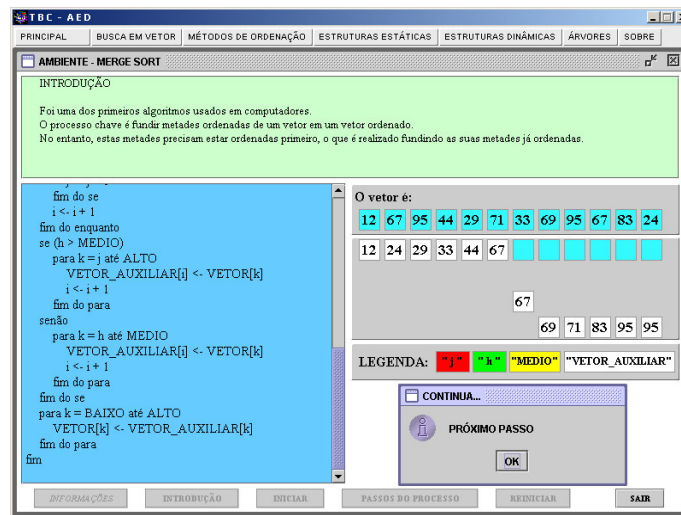


Figura 4. Tela *Merge Sort* com janela de mensagem interna

Uma característica importante do *TBC-AED* é ser auto-explicativo, bastando deslizar com o *mouse* sobre as partes da tela para ver uma breve descrição sobre a região posicionada. Isso deixa o usuário mais à vontade e sem preocupações quanto a quaisquer peculiaridades, também por se tratar de um produto de *software* direcionado para alunos iniciantes em cursos de Computação.

Dentre as vantagens da abordagem construtiva, estão mecanismos para facilitar o processo de abstração, o fato de a animação refletir a interação com o aprendiz e as várias facilidades para a detecção visual de erros. Com isso, incentiva-se o processo de compreensão e autocorreção [Garcia *et al.* 1997]. Isso pode ser facilmente percebido em processos recursivos, os quais são de difícil explicação teórica, mas que podem ser vistos facilmente através da animação gráfica disponível em ambientes que utilizam algum algoritmo desse tipo (como *merge sort* e *quick sort*).

Com isso, verifica-se que a organização do *TBC-AED* é amplamente didática e esse fato serve de grande utilidade para o ensino de disciplinas que apresentam como ementa os tópicos relacionados. Além disso, é uma experiência desafiadora de graduandos, mestres e doutores tornar o ensino de Computação mais dinâmico, ao

despertar o seu interesse a terem a mesma iniciativa, se possível com igual dedicação e empenho, para melhorar e aprimorar a formação de recursos humanos para a tecnologia.

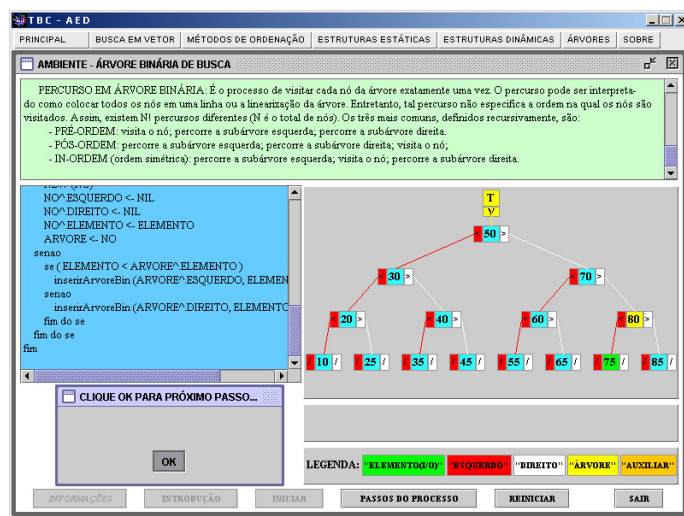


Figura 5. Tela *Árvore Binária* com suas peculiaridades

Como consequência da evolução, intrínseca a qualquer projeto, o *TBC-AED* foi disponibilizado via *Web*, através de *applets*¹, gerando o *TBC-AED/WEB* (**T**reinamento **B**aseado em **C**omputador para **A**lgoritmos e **E**struturas de **D**ados via **W**eb) para que possa ter maior acessibilidade, além de possibilitar a troca de idéias, através da interatividade com usuários diversos. A idéia de criar o *TBC-AED/WEB* surgiu para evitar que o estudante não precise fazer o *download* do *TBC-AED* para estudar, bastando ter uma versão do aplicativo apenas em sua estação de trabalho em casa, para trabalhar *offline*. Quando estiver em alguma estação conectada à *Internet*, ele pode executar o programa diretamente no navegador, desde que esta estação possua a *JVM* (*Java Virtual Machine*) instalada.

Quanto às características relacionadas à funcionalidade e à aparência, o *TBC-AED/WEB* não difere do *TBC-AED*. Dessa forma, a partir do acesso *Web*, espera-se gerar uma interatividade maior ainda com outros pesquisadores, difusão da pesquisa, contribuição à comunidade científica da área e *feedback* em relação ao trabalho realizado.

Assim sendo, o *TBC-AED* e o *TBC-AED/WEB* é uma ferramenta valiosa para a formação de alunos, pois atende a disciplinas que estão no início do curso e que são pré-requisitos necessários para as disciplinas mais específicas de períodos avançados. Quando feita uma boa base, o rendimento e o desempenho aumentam, proporcionando melhores resultados, melhores currículos e melhores profissionais para o mercado.

6. *TBC-AED* em Uso

Alguns seminários de apresentação e simulação de recursos disponíveis no programa foram realizados durante o primeiro semestre de 2005 aos alunos que cursavam a disciplina e aos alunos que a utilizariam no segundo semestre, agora com a aplicação contínua da ferramenta.

¹ *Applets* são programas Java que podem ser embutidos em documentos HTML. Quando um navegador carrega uma página da *Web* que contém um applet, o applet é executado ([DEITEL, 2003]).

Até o momento, o uso da ferramenta atende às expectativas, pois foi notado que os alunos ficaram mais motivados e mais atentos, uma vez que estão utilizando o computador diretamente. Além disso, a ferramenta foi disponibilizada aos alunos, os quais podem acessar exemplos práticos extraclasse, ou seja, em casa ou em outro lugar que não seja durante o horário da aula.

Foi observado que a ferramenta altera o ânimo dos alunos, devido ao fato de que conceitos abstratos passam a ser melhor visualizados e compreendidos, melhorando a dinamização do ensino. Isso também contribuiu para um tempo maior reservado à resolução de exercícios, levando à sedimentação da teoria. Isso resulta em maior expectativa do aumento do índice de aprovação em uma disciplina importante para a Computação.

A partir da apresentação e disponibilização da ferramenta aos alunos, aqueles que a utilizaram obtiveram êxito na disciplina, perfazendo um total geral de aprovação superior àqueles que não a utilizaram. Devido a esse fato, para melhorar os resultados, optou-se por dividir a carga horária da disciplina, antes apenas teórica, em aulas expositivas (teóricas) e de laboratório (uso da ferramenta, incluindo exercícios de programação) para o segundo semestre de 2005.

7. Conclusões e Trabalhos Futuros

Sob todo conjunto de informações obtidas até o presente momento, pode-se perceber que existe uma gama de recursos que possibilitam o desenvolvimento de ferramentas computacionais para o auxílio da educação superior em Computação, com destaque para a linguagem de programação *Java*, que conta com a portabilidade. Isso faz com que as novas tecnologias atinjam mais rapidamente seu público alvo.

Além disso, observou-se que o uso de produtos de *software* para o ensino de programação e algoritmos é uma idéia interessante, se for amadurecida e desenvolvida de forma cuidadosa e estruturada. Isso fornece novas experiências para professores e alunos, que se proponham a trabalhar com esse tema. Além disso, eles terão sua qualidade de ensino elevada e poderão ser capazes de avançar mais rapidamente no campo do conhecimento, através da agilidade de processos didáticos e da participação ativa do docente da área.

Conclui-se também que uma boa elaboração de produtos tecnológicos que facilitem a transmissão de conhecimentos em Computação deve ser acompanhada de expressiva pesquisa no campo de novas metodologias de ensino. A principal finalidade disso é cada vez mais haver contribuições que incorram no aprimoramento do ensino superior e na formação de profissionais melhor qualificados para o mercado. Isso reflete principalmente sobre futuros professores, uma vez que aqueles que têm passado por esse tipo de experiência durante a graduação terão grande interesse em executar o mesmo processo quando estiverem lecionando.

Este trabalho deixa como contribuição o *TBC-AED* e o *TBC-AED/WEB* que englobam temas relacionados a algoritmos e estruturas de dados. Além disso, faz uma revisão da literatura envolvendo a importância de um bom ensino de algoritmos e programação e tópicos de Informática na Educação. Isso estimula mudanças na educação em cursos que formam profissionais para a área tecnológica e incentivam docentes a melhorarem sua forma de pesquisa e ensino envolvendo Computação e Informática.

Com relação aos primeiros resultados da aplicação da ferramenta, estão sendo notadas maior atenção e facilidade de aprendizagem. Dessa forma, o sistema continuará a ser utilizado ao longo das aulas ministradas para a disciplina Algoritmos e Estruturas de Dados II, oferecida pelo DCC/UFLA, incluindo questionários e análise dos resultados das avaliações.

Com o *TBC-AED/WEB*, pretende-se alcançar maior acessibilidade e troca de idéias, dada a interatividade com usuários diversos. A partir disso, também é interessante atacar outro assunto de grande importância, a teoria dos grafos, a fim de ampliar o conhecimento e contribuir para a melhoria do ensino na área, através de novas ferramentas para o ensino de Computação.

Referências

- Almeida, M. E. (2000) “Informática e Formação de Professores”, Série de Estudos – Educação à Distância (ProInfo). Ministério da Educação. Secretária de Educação à Distância. v. 2
- Almeida, M. S. e Nogueira, C. R. D. (2001) “A *Internet* no Trabalho Pedagógico dos Professores”, Informática na Educação – Artigos Científicos. Universidade Regional Integrada do Alto Uruguai e das Missões – URI – RS.
- Alves, A. A. e Costa, M. R. (2004) “*algoPUC*: Uma Ferramenta de Apoio à Programação para Iniciantes”, III Workshop de Educação em Computação e Informática do estado de Minas Gerais (WEIMIG’ 2004). Belo Horizonte, MG, Brasil.
- Azeredo, P. A. (2000) “Uma proposta de Plano Pedagógico para a Matéria de Programação”, Anais do II Curso: Qualidade de Cursos de Graduação da Área de Computação e Informática (WEI 2000). Editora Universitária Champagnat.
- Azul, A. A. e Mendes, A. J. “EDDL: Um Programa Didático sobre Estruturas de Dados Dinâmicas Lineares”, 3º Simpósio Investigação e Desenvolvimento de *Software* Educativo – 1998. Évora, Portugal. Disponível em: <http://www.minerva.uevora.pt/simposio/comunicacoes/Mendes/Eddl.html>. Acesso: 15 jun 2005.
- Black, P. E. “Dictionary of Algorithms and Data Structures”. NIST (*National Institute of Standards and Technology*). Disponível em <http://www.nist.gov/dads/>. Acesso: 15 jun 2005.
- Borges, K. S. e Filho, H. B. R. (2005) “A Importância dos Grupos de Estudos na Formação Acadêmica”. XIII Workshop de Educação em Computação (WEI’2005). São Leopoldo, RS, Brasil.
- Faria, E. S. J. e Coello, J. M. A. (2005) “Um Estudo Empírico dos Efeitos do Uso de Trabalho Colaborativo no Aprendizado de Programação em Cursos de Graduação em Computação”. XIII Workshop de Educação em Computação (WEI’2005). São Leopoldo, RS, Brasil.
- Faria, E. S. J., Vilela, J. M. e Coello, J. M. A. (2005) “Um Sistema de Aprendizado Colaborativo de Programação Baseado em Agentes chamado *Learn In Group*”. XIII Workshop de Educação em Computação (WEI’2005). São Leopoldo, RS, Brasil.
- Flávio, D. “TED - Tutorial de Estruturas de Dados”, desenvolvido durante estágio supervisionado na Universidade do Vale do Itajaí - 2004. Disponível em <http://www.tutorialdeestruturadedados.8m.com/>. Acesso: 20 jun 2005.

- Garcia, I. C., Rezende, P. J. e Calheiros, F. C. (1997) “Astral: Um Ambiente para Ensino de Estruturas de Dados através de Animações de Algoritmos”, Revista Brasileira de Informática na Educação (RBIE’1997) nº 01, <http://www.inf.ufsc.br/sbc-ie/revista/nr1/garcia.htm>. Ambiente ASTRAL disponível em: <http://www.dcc.unicamp.br/~rezende/ASTRAL/>. Acesso: 28 abr 2005.
- Giraffa, L., Marczak, S. e Prikladnicki, R. (2005) “PSD-E: Em Direção a um Processo para Desenvolvimento de *Software* Educacional”. XI Workshop de Informática na Escola (WIE’2005). São Leopoldo, RS, Brasil.
- Júnior, J. C. R. P. e Rapkiewicz, C. E. (2004) “O Processo de Ensino e Aprendizagem de Algoritmos e Programação: Uma Visão Crítica da Literatura”, III Workshop de Educação em Computação e Informática do estado de Minas Gerais (WEIMIG’ 2004). Belo Horizonte, MG, Brasil.
- Júnior, J. C. R. P., Rapkiewicz, C. E., Delgado, C. e Xexeo, J. A. M. (2005) “Ensino de Algoritmos e Programação: Uma Experiência no Nível Médio”. XIII Workshop de Educação em Computação (WEI’2005). São Leopoldo, RS, Brasil.
- Koslowski, S. R. e Nunes, M. A. S. N. (2001) “O Computador como Ferramenta Pedagógica nas Séries Iniciais”, Informática na Educação – Artigos Científicos. Universidade Regional Integrada do Alto Uruguai e das Missões – URI – RS.
- Raabe, A. L. A. e Silva, J. M. C. (2005) “Um Ambiente para Atendimento das Dificuldades de Aprendizagem de Algoritmos”. XIII Workshop de Educação em Computação (WEI’2005). São Leopoldo, RS, Brasil.
- Setúbal, J. C. (2000) “Uma proposta de Plano Pedagógico para a Matéria de Computação e Algoritmos”, Anais do II Curso: Qualidade de Cursos de Graduação da Área de Computação e Informática (WEI 2000). Editora Universitária Champagnat.
- Soares, T. C. A. P., Cordeiro E. S., Stefani Í. G. A., Tirelo, F. (2004) “Uma Proposta Metodológica para o Aprendizado de Algoritmos em Grafos Via Animação Não-Intrusiva de Algoritmos”, III Workshop de Educação em Computação e Informática do Estado de Minas Gerais (WEIMIG’ 2004). Belo Horizonte, MG, Brasil.
- Souza, P. C. e Wazlawick, R. S. (1997) “Ferramenta de Autoria para a Criação de Ambientes Construtivistas em Realidade Virtual”. Anais do VIII Simpósio Brasileiro de Informática na Educação. Instituto Tecnológico de Aeronáutica (ITA), v. 1.
- Wirth, N. (1989) “Algoritmos e Estruturas de Dados”. LTC Informática-Programação.
- Yamamoto, F. S., Silva, A. F., Zanutto, J. e Zampirolli, F. A. (2005) “Interdisciplinaridade no Ensino de Computação”. XIII Workshop de Educação em Computação (WEI’2005). São Leopoldo, RS, Brasil.
- Zambalde, A. L. e Alves, R. M. (2002) “Introdução à Informática Educativa”, Textos Acadêmicos para o Curso de Pós-Graduação ‘*Lato Sensu*’ (Especialização) à Distância – Informática em Educação. UFLA/FAEPE.