

UMA CONTRIBUIÇÃO METODOLÓGICA PARA O ENSINO DE RECURSIVIDADE EM CURSOS DE CIÊNCIA DA COMPUTAÇÃO

09/2011

Novas Tecnologias em Educação

Instituto Federal de Educação Ciência e Tecnologia do Pará (IFPA)
MARQUES, Andrea Lilian
alilian.costa@gmail.com

Universidade Federal do Pará (UFPA)
FERREIRA, Benedito Pinheiro
ferreira@ufpa.br

Universidade Federal do Pará (UFPA)
JUNIOR, Luiz Dourado
ldourado1980@globo.com

Introdução

Muitos estudos, sob diferentes enfoques, têm trazido para discussão as dificuldades inerentes ao processo de ensino e aprendizagem de disciplinas relacionadas à programação (Pereira e Rapkiewicz 2004, Powers e Gross 2006, Byckling e Sajaniemi 2006, Bubb, 2006, Santos e Costa, 2006, Rocha et al., 2010). O que explica esse interesse é a importância crucial que tal atividade de ensino exerce em um curso de graduação de computação e informática, aliada ao reconhecimento da complexidade de tal processo, que envolve muito mais do que simples transferência/aquisição de informações. De fato, a prática de ensino de programação oferece constantes desafios, pois exercê-la de forma eficaz exige o desenvolvimento no aluno da capacidade de mobilizar um conjunto de ferramentas simbólicas que representam ações computacionais para a resolução de problemas de natureza variada. Na prática pedagógica como já destacado por Raabe e Silva (2005), um dos fatores que tem contribuído para o desenvolvimento de dificuldades dos alunos são as exigências de raciocínio matemático-lógico para a resolução de problemas e conceitos abstratos

típicos das disciplinas que envolvem programação. Segundo Santos e Costa (2006) a busca por melhores resultados no processo de ensino e aprendizagem nessa área, torna indispensável a necessidade constante de pesquisas que visem a atualizações das didáticas de ensino de forma geral. O objetivo disso é transformar processos abstratos em concretos, por meio de produtos de softwares que possibilitem melhor compreensão dos tópicos abordados e maior interação entre aluno-objeto de trabalho (neste caso, o computador).

Para Soares et. al. (2004) a partir de observações dentro de disciplinas de graduação, percebeu-se um melhor resultado no aprendizado quando da aplicação de atividades práticas de desenvolvimento de simuladores e ferramentas visuais didáticas de representação de conceitos abstratos. Além disso, acompanhando esse desenvolvimento, contribuições advindas de outros pesquisadores têm fornecido novos instrumentos que podem melhorar a compreensão de determinados conteúdos. Isso promove a ampliação do acervo existente, a melhor organização de currículos para o mercado de trabalho e o acesso facilitado de tais recursos por meio do suporte web.

No que tange às disciplinas relacionadas à programação, o tema recursividade é uma técnica essencial para o amadurecimento da capacidade de programação do estudante de computação. Essa técnica permite a codificação de problemas complexos de forma relativamente simples, pelo alto grau de abstração, ficando transparentes operações como as de empilhamento/desempilhamento e ativações subsequentes de um módulo. Isso torna o desenvolvimento da solução mais simples, bem como seu entendimento. Cabe destacar que o aprendizado da técnica de recursividade muitas vezes está envolvido com dificuldades, pois para seu entendimento é exigido do aluno um certo grau de abstração. Entretanto, pode-se trabalhar com a hipótese de que um aspecto importante para se facilitar essa aprendizagem será favorecer a compreensão, especialmente por meio de exemplos simples, das operações subjacentes a um algoritmo recursivo (WEISS, 2003). Em um levantamento não exaustivo, identificou-se escassez do uso de tecnologias já existentes com intuito de abordar o tema, especialmente, no que tange à simulação passo a passo da execução de algoritmos recursivos.

Objetivo

O objetivo desta pesquisa foi desenvolver, a partir das tecnologias de informação e comunicação, particularmente técnicas de animação, um recurso de apoio ao ensino de recursividade ministrada na disciplina de estrutura de dados em cursos de graduação de computação.

Metodologia

Tecnicamente, trata-se de uma animação em formato “SWF”, do Flash, construída em duas etapas, que empregam duas ferramentas computacionais: a) Editor de slides – Power Point; b) Conversor de Apresentações - Ispring.

Etapa 1 - Criação dos slides passo a passo

Cada um dos slides foi dividido de forma a conter do lado esquerdo o código da função recursiva e do outro, simultaneamente, o processo em execução na memória do computador.

O modo de funcionamento é: o código visualizado do lado esquerdo da tela do slide continha uma tarja cinza sobre a linha de código a ser executada, sendo possível visualizar os resultados dessa execução no esquema representativo da memória, à direita do slide. Esse procedimento foi utilizado até a última linha de comando do código, encerrando-se com a apresentação do resultado obtido pela execução final da função recursiva.

Etapa 2 - Conversão do arquivo ppt ou pps para swf

Após a criação dos slides utilizou-se um conversor de apresentação – Ispring. Essa ferramenta foi escolhida, justamente, pela possibilidade de converter apresentações em formato “pps” e “ppt” para o formato utilizado pelo Flash; pela facilidade de uso, bem como pelo fato deste recurso estar disponível gratuitamente na internet. Uma vez baixado e instalado o programa Ispring, um botão denominado Flash será visualizado na barra de ferramentas do editor de slides. É necessário abrir o arquivo contendo a apresentação e posteriormente clicar sobre o botão Flash. Uma janela será aberta solicitando que o usuário informe o nome do arquivo a ser convertido para a extensão .swf; feito isso a conversão será feita automaticamente.

Resultados

Em termos funcionais, o recurso permite que o aluno visualize todos os eventos de processamento subjacentes à abstração de recursividade, simultaneamente à evolução do controle, passo a passo, pelas linhas de código. Na figura 1 (em anexo) pode-se observar as quatro primeiras telas capturas da sequência de apresentação dos slides.

Pedagogicamente, a ferramenta enfatiza a animação gráfica, que pode favorecer a aprendizagem, bem como facilitar a apresentação de conceitos abstratos, tais quais os envolvidos no tema alvo do recurso descrito (Santos e Costa, 2006). Além disso, o recurso educacional foi concebido de forma a se adequar ao andamento individual de compreensão de cada aluno, pois ele próprio pode ir avançando nas telas de acordo com o seu ritmo de compreensão de cada passo do algoritmo em estudo, na figura 2 (em anexo), tem-se uma ilustração do recurso em sua fase final.

O uso do arquivo na extensão .swf facilita o armazenamento, a distribuição e integração do recurso educacional a um ambiente virtual de ensino e aprendizagem. Isso provê o alcance de uma característica desejável, segundo Santos e Costa (2006), que é a de tornar o recurso acessível ao aluno fora do ambiente escolar, de modo que esse possa ser consultado livremente nos momentos em que eventuais dúvidas possam surgir. Além disso, cabe destacar a vantagem de se colocarem todos os recursos em um único ambiente, evitando-se mudanças frequentes de janelas.

Por meio do recurso desenvolvido é possível ter uma melhor gestão do tempo da aula. O tempo, que seria gasto em transcrever explicações do quadro negro para o caderno, pode ser utilizado para resolução de exercícios de aplicação e fixação. Assim, pode-se alcançar maior interação entre o professor e os seus alunos, dada a ampliação do tempo disponível para questionamentos, contribuindo-se dessa forma para uma maior colaboração e diálogo entre os participantes do processo, essencial à aprendizagem, aspecto especialmente valorizado por Freire (2005).

Além da possibilidade de uso pelo aluno, enquanto realiza seus estudos individuais e simulações acerca do tema, o recurso ora apresentado, pode ser utilizado

pelo professor por ocasião das aulas para exemplificar/explicar o assunto recursividade; acrescenta-se que conforme a opção tecnológica, o recurso pode também ser disponibilizado para consultas futuras pelos alunos, o que pode ser feito tanto em um ambiente virtual de aprendizagem, como em uma página web da disciplina.

Conclusões

Pôde-se perceber, no desenvolvimento deste trabalho, que existe uma elevada quantidade de recursos que podem facilitar o desenvolvimento de ferramentas computacionais para o enfrentamento concreto das freqüentes dificuldades encontradas no processo de ensino e aprendizagem nas disciplinas de programação.

Dentre estes recursos existentes, cabe destaque cada vez maior ao uso de produtos de software que, por esta razão, precisam ser levados em consideração nos processos pedagógicos, de modo a serem inseridos de maneira bem estruturada e planejada, bem como favorável a uma facilitada apreensão de conteúdos, muitos dos quais acabam se tornando áridos sem esses recursos. Com isso, os professores poderão ter a qualidade de seu trabalho elevada e os estudantes poderão ser capazes de avançar mais rapidamente em seu campo de conhecimento, por meio da agilidade de processos didáticos.

Conclui-se também que uma boa elaboração de produtos tecnológicos que facilitem a apreensão de conhecimentos em computação deve ser acompanhada de pesquisas que gerem contribuições teóricas e metodológicas importantes com implicações educacionais. Assim, entende-se que o objetivo do artigo, constituído a partir da escassez de recurso que abordem o tema recursividade, foi alcançado a partir de orientações pedagógicas consistentes, especialmente, as que enfatizam o aspecto da simulação e da colaboração entre os alunos, e que podem trazer uma contribuição ao ensino do tema nos cursos superiores em computação.

Uma limitação deste trabalho diz respeito a falta de um estudo mais sistematizado com relação a averiguação do nível de aprendizagem facilitada e/ou suscitada em turmas de programação por meio do recurso aqui apresentado. Propomos

como trabalho futuro um estudo que apresente os resultados da aplicação e uso deste recurso em sala de aula e sua importância para a aprendizagem da recursividade pelos alunos. Esse tipo de aplicação poderá confirmar o potencial que julgamos pertencer ao recurso. Certamente um estudo dessa natureza será um importante passo para o avanço no conhecimento nesse tema específico e os resultados poderão contribuir para o aperfeiçoamento do trabalho realizado.

Referências

BUBB, T. A. **An Active Learning Approach to Teaching the Data Structures Course**. In: technical Symposium on Computer Science Education – SIGCSE. Houston, Texas USA, march 1-5, 2006.

BYCKLING, P.; SAJANIEMI, J. **Roles of Variables and Programming Skills Improvement**. In: Technical Symposium on Computer Science Education – SIGCSE. Houston, Texas USA, March 1-5, 2006.

FREIRE, P. **Pedagogia do oprimido**. 44 Ed, Paz e Terra, 2005.

PEREIRA JUNIOR, J. C. R., RAPKIEWICZ C. **O processo de ensino e aprendizagem de fundamentos de programação: Uma Visão Crítica da Pesquisa no Brasil**, WEI RJES, 2004.

POWERS E GROSS. **Tools for Teaching Introductory Programming: What Works?** In: Technical Symposium on Computer Science Education – SIGCSE 2006. Houston, Texas USA, March 1-5.

RAABE, A. L. A., SILVA, J. M. C. (ANO) **Um Ambiente para Atendimento as Dificuldades de Aprendizagem de Algoritmos**. In XXV Congresso da Sociedade Brasileira de Computação. São Leopoldo/RS, 2005.

ROCHA, P. S., FERREIRA B. P., MONTEIRO D., NUNES, D. da S., GOES, H. C. r do N. **Ensino e Aprendizagem de Programação: Análise da Aplicação de Proposta Metodológica Baseada no Sistema Personalizado de Ensino**, RENOTE. Revista Novas Tecnologias na Educação, v. 8, p. 1-11, 2010.

SANTOS, R. P. e COSTA, H. A. X. **Análise de Metodologias e Ambientes de Ensino para Algoritmos, Estruturas de Dados e Programação aos iniciantes em Computação e Informática.** In INFOCOMP – JOURNAL OF COMPUTER SCIENCE, Lavras/MG – Brasil, v. 5, n. 1, p. 41-50, 2006.

WEISS, M. A. **Recursion.** In: Data Structures e Problem Solving Using C++. 2nd Edition. Pearson Education International. p. 265-319, 2003.

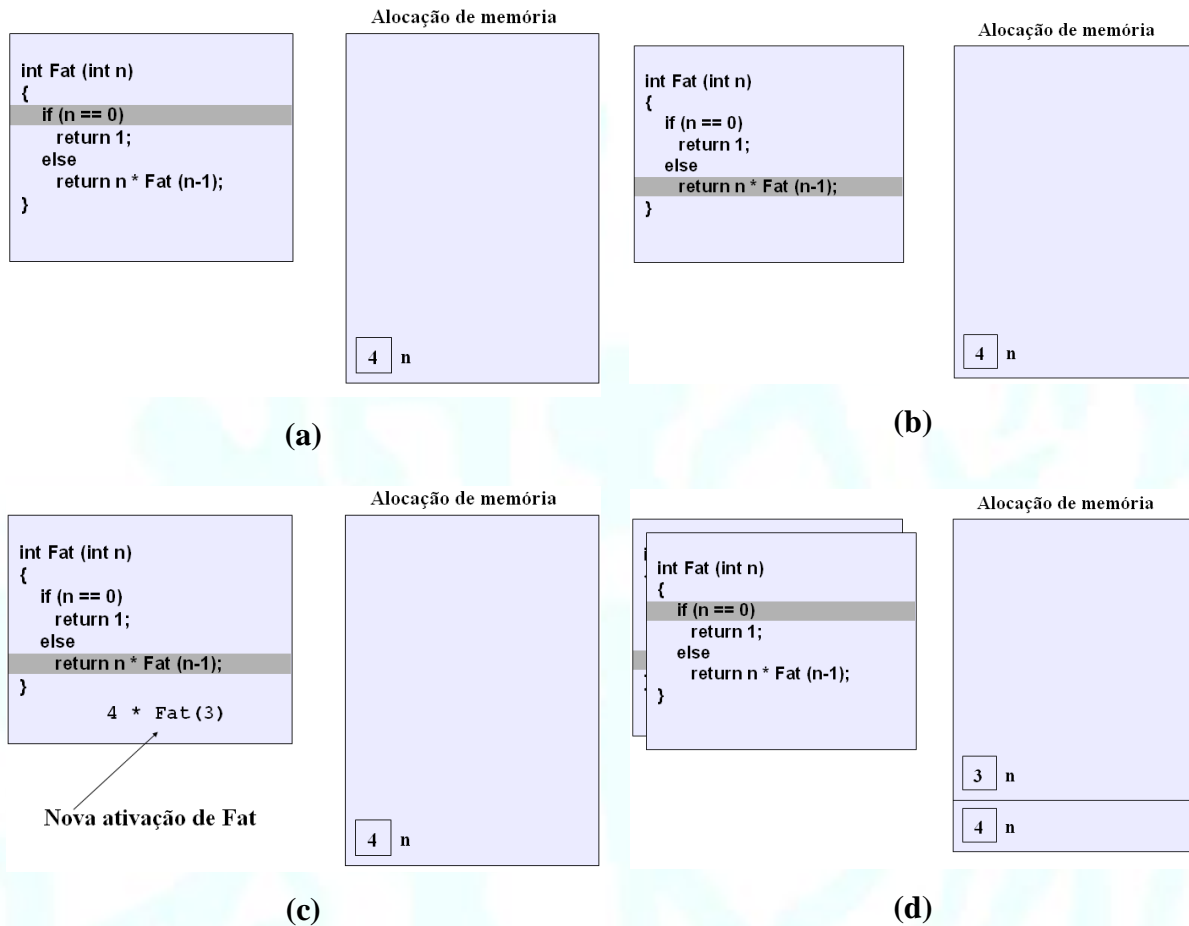


Figura 1: Passo a passo dos quatro primeiros slides da apresentação

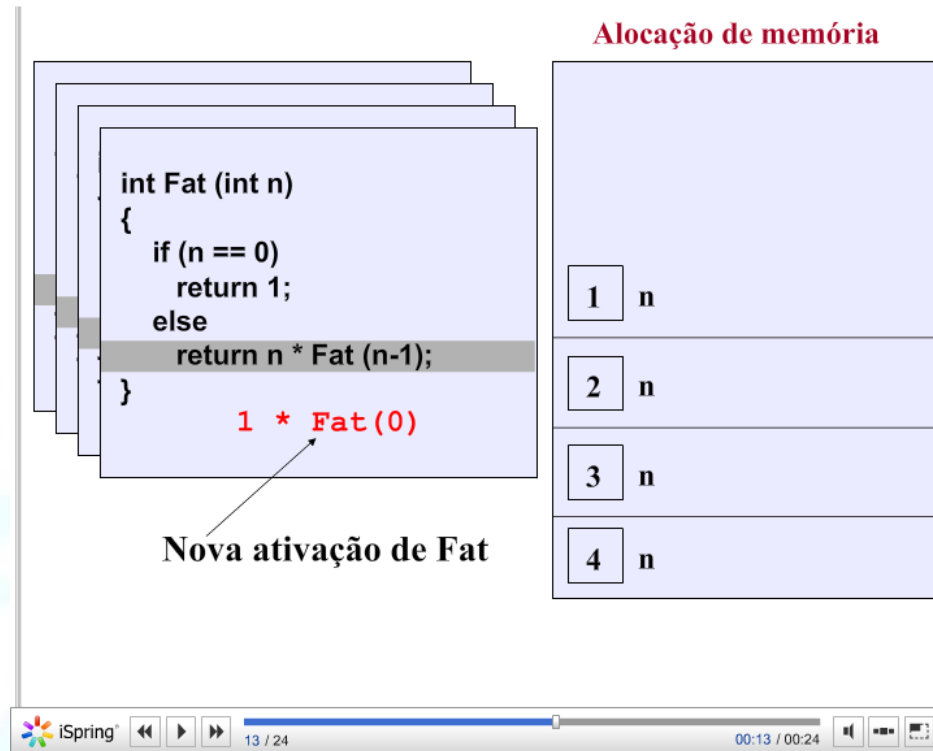


Figura 2: Tela de execução do recurso educacional após conversão à versão flash.