

Objetos de Aprendizagem para o Apoio ao Ensino da Linguagem de Programação C

Cristian Cechinel, Alex Veloso da Silveira, Rafael Silveira da Silveira, Érico de Moraes Nunes, Alexandre Amaral Moreira, Giovani Bolzan Cogo, Carlos Michel Betemps, Reginaldo da Nóbrega Tavares

Universidade Federal do Pampa, UNIPAMPA/Bagé, Rua Carlos Barbosa S/Nº
Bairro Getúlio Vargas, 96412-420, Bagé, RS, Brasil

{cristian.cechinel, alexvs87, rafaelss.svp, nunes.eric,
alexandream.ec, giovanibcogo, carlos.betemps, regi.ntavares}@gmail.com

***Abstract.** This work presents the very first stage of a research project which aims to develop learning objects to support algorithms and programming teaching and learning process. The paper describes the developed learning objects (focused on C programming language), their contents and goals, and the evaluation of these materials by the students. This work also highlights some student's difficulties on learning how to program, and points out which are the didactic materials considered relevant by them to support their studies.*

***Keywords:** Learning Objects, Algorithms and Programming.*

***Resumo** Este trabalho apresenta a primeira etapa de execução de um projeto voltado para o desenvolvimento de objetos de aprendizagem para o apoio ao ensino de Algoritmos e Programação. O trabalho descreve os primeiros objetos de aprendizagem desenvolvidos (para o ensino da linguagem de programação C), seus conteúdos e objetivos, e uma avaliação inicial dos acadêmicos sobre o uso dos mesmos. O artigo também apresenta algumas observações sobre as dificuldades enfrentadas pelos estudantes no processo de ensino-aprendizagem dos conteúdos de programação e quais materiais didáticos os alunos apontam como úteis para o apoio a esse processo.*

***Palavras-chave:** Objetos de Aprendizagem, Algoritmos e Programação.*

1. Introdução

O presente trabalho tem como objetivo principal relatar as atividades e resultados obtidos na primeira etapa de execução do projeto intitulado “**Desenvolvimento de um Repositório de Objetos de Aprendizagem para o Apoio ao Ensino dos Fundamentos da Área de Conhecimento de Algoritmos e Programação e Estruturas de Dados**” que está sendo desenvolvido por professores e acadêmicos do curso de Engenharia de Computação da Universidade Federal do Pampa (Unipampa) – Campus de Bagé. O projeto tem como objetivo auxiliar os alunos da disciplina de Algoritmos e Programação (AP) através da criação de objetos de aprendizagem (OAs), que neste primeiro momento estão sendo desenvolvidos em formato de vídeos explicativos (tutoriais). Esses tutoriais são desenvolvidos a partir de gravações do *desktop* do computador e contêm informações e dicas sobre como utilizar e programar

na linguagem C algumas das principais estruturas de programação estudadas nas disciplinas de AP. O artigo está subdividido da seguinte forma: nessa seção é apresentada uma pequena introdução sobre o trabalho; na caracterização do problema são descritas algumas das dificuldades encontradas no processo de ensino-aprendizagem de AP e as motivações para o desenvolvimento dos OAs aqui apresentados; a seção 3 apresenta conceitos e idéias sobre OAs e sobre repositórios de OAs; a seção 4 fala sobre o processo desenvolvimento dos OAs em questão, as ferramentas utilizadas na construção dos mesmos e os conteúdos abordados; na seção é descrito o levantamento de dados realizado junto aos acadêmicos; finalmente, o artigo relata os primeiros resultados obtidos por esse projeto, e as considerações finais.

2. Caracterização do Problema

O conceito de algoritmos é baseado na capacidade de desenvolver soluções através de instruções obedecendo a uma seqüência lógica. As disciplinas de AP têm como objetivo dar um passo introdutório à programação e estão normalmente vinculadas aos currículos dos cursos de engenharia e da área de computação. Na Unipampa – Campus de Bagé, essa disciplina é oferecida para 6 (seis) cursos diferentes – Engenharias de Computação, Química, Alimentos, Produção, Energias Renováveis e Ambiente; e Licenciatura em Física – e apresenta uma das maiores taxas de reprovação e desistência dentro da universidade (acima de 50%).

As dificuldades inerentes ao processo de ensino-aprendizagem de AP estão relacionadas a diversos fatores e são objeto de estudo em vários trabalhos. Por exemplo: Smith (1981) aponta as dificuldades que os acadêmicos encontram em relacionar os significados das informações recebidas e as suas dependências com outros conteúdos; Falkembach *et al.*(2003) relatam a existência de dificuldades no entendimento dos enunciados dos problemas, e Olsen (2005) fala sobre a própria falta de habilidade na construção de solução para os problemas apresentados. Um levantamento informal realizado junto aos acadêmicos da Unipampa mostrou que os alunos vinculam o fracasso nas disciplinas de algoritmos à deficiência apresentada no currículo escolar médio no que se refere à existência de uma matéria que esteja vinculada ao desenvolvimento do raciocínio lógico.

Com o intuito de amenizar as dificuldades enfrentadas pelos acadêmicos, diversos grupos vêm trabalhando diretamente na realização de experimentos práticos, e na construção de ferramentas e metodologias para facilitar o processo de ensino-aprendizagem dessa área, sempre direcionando seus trabalhos para as principais dificuldades levantadas junto aos seus estudantes. Dentre as inúmeras iniciativas existentes, pode-se citar: 1) ASTRAL -Ambiente para Ensino de Estruturas de Dados através de Animações de Algoritmos [Garcia *et al.* 1997]; 2) o ambiente TBC-AED - Treinamento Baseado em Computador para Algoritmos e Estruturas de Dados [Dos Santos e Costa 2005], e 3) ALICE - Ambiente para Atendimento as Dificuldades de Aprendizagem de Algoritmos [Raabe e Silva 2005].

O aumento expressivo da quantidade de ferramentas computacionais para o apoio ao ensino e o rápido avanço da internet impulsionaram a criação de mecanismos capazes de organizar e conectar esses conjuntos de materiais didático-pedagógicos, e transformá-los em blocos de conhecimento passíveis de reutilização e compartilhamento, os chamados

Objetos de Aprendizagem e Sistemas de Gerenciamento de Aprendizagem [Downes 2001]. Nesse sentido, um grupo de acadêmicos e professores do curso de Engenharia de Computação da Unipampa – Campus de Bagé está desenvolvendo um repositório de objetos de aprendizagem para o apoio ao processo de ensino-aprendizagem dos fundamentos de AP.

3. Objetos de Aprendizagem e Repositórios de Objetos de Aprendizagem

De acordo com L’Allier (1997) um Objeto de Aprendizagem (OA) é definido como “a menor estrutura instrucional independente que contém um Objetivo, uma Atividade de Aprendizagem e uma Avaliação”. Nesse sentido, um OA consiste em um componente digital, base de um curso, unidade ou lição que pode ser reutilizado para criar outras estruturas instrucionais únicas. Algumas das principais características apresentadas pelos OAs que estimulam a sua utilização e disseminação na internet são descritas por Longmire (2000): 1) Flexibilidade – desenvolvido para ser reutilizado, não necessita ser reescrito para cada novo contexto; 2) Personalização – facilita a personalização de conteúdos permitindo a recombinação de materiais; 3) Interoperabilidade – permite a definição de especificações de design, de desenvolvimento e de apresentação baseadas em necessidades da organização e mantém a capacidade de operar em conjunto com outros sistemas de aprendizagem e contextos; 4) Aumento de significância do conhecimento – com a reutilização de um OA várias vezes e em diferentes situações, seu conteúdo é consolidado naturalmente à medida que o tempo passa.

Além dos próprios OAs e de uma descrição padronizada por meio de uma metalinguagem apropriada (que tem como objetivo facilitar a atualização, a busca e a seleção dos mesmos), existe ainda um outro componente necessário para uma utilização adequada dos OAs na educação; esse componente é chamado de Sistema de Gerenciamento de Conteúdos de Aprendizagem. De acordo com Brennan *et al.* (2001), um Sistema de Gerenciamento de Conteúdos de Aprendizagem é utilizado para “criar, armazenar, agrupar e disponibilizar conteúdos personalizados de e-learning na forma de OA”. De modo geral os Sistemas de Gerenciamento de Conteúdos de Aprendizagem são formados pelos seguintes módulos: Repositório de OAs – base de dados central onde os conteúdos de aprendizagem são armazenados e gerenciados; Aplicação Automatizada para Autoria – ferramenta utilizada para criar os OAs que estão acessíveis no repositório; Interface Dinâmica para Endereçamento e Entrega – utilizada para apresentar um objeto de acordo com o perfil do usuário, provendo links para fontes selecionadas de informação e feedback sempre que necessário; e uma Aplicação Administrativa – responsável pelo registro dos estudantes, por dar início a cursos com base em catálogos e prover relatórios sobre o progresso dos usuários.

4. Os Objetos de Aprendizagem Desenvolvidos

Nessa primeira etapa de execução do projeto foram desenvolvidos OAs para o apoio ao ensino da linguagem de programação C (linguagem adotada nas disciplinas de AP da instituição). Esses OAs possuem o formato de tutoriais (vídeos em flash) e foram criados a partir de gravações do *desktop* do computador enquanto algum determinado problema estava sendo programado.

4.1. Ferramentas Utilizadas

Para realizar a gravação dos vídeos foi utilizada a ferramenta Wink (www.debugmode.com/wink). O Wink é um software editor e gravador de vídeos que gera um arquivo em formato flash (.swf) e mostra tudo o que o usuário executou na tela durante o período de gravação. Dentre os motivos pelos quais o Wink foi adotado estão: 1) é disponibilizado gratuitamente para *download*; 2) possibilita fácil edição dos vídeos gravados com recursos como balões de texto e botões indicadores; e 3) gera um arquivo de tamanho pequeno e em formato amplamente utilizado, ideal para ser disponibilizado na internet.

Os programas que foram temas dos OAs foram desenvolvidos utilizando o editor Kate - editor de texto de código aberto – rodando sobre uma plataforma Linux. O Kate possui um terminal integrado, o que possibilitou a edição, a compilação e a execução dos programas dentro de uma mesma janela. O compilador C utilizado foi o GCC (Gnu Compiler Collection). Pelo fato de muitos acadêmicos não terem intimidade com o uso do Linux, também foi desenvolvido um OA específico para apresentar a IDE DevC++ rodando no sistema operacional Windows.

4.2. Metodologia de Desenvolvimento

Os OA foram desenvolvidos com base em um planejamento prévio que era elaborado pelos professores das disciplinas de AP e repassados aos acadêmicos envolvidos no projeto. Para uniformizar o processo de construção dos tutoriais, foram estabelecidos padrões a serem respeitados com relação a: 1) informações que deveriam ser inseridas na tela de apresentação e na tela de encerramento; 2) cor, fonte, formato e posições dos balões de texto e botões de ação; 3) tamanho máximo em kbytes dos arquivos gerados; 4) editor que deveria ser utilizado para programação; 5) tamanho da janela para gravação; e 6) fonte e indentação do código C programado. Além dos padrões definidos para todos os OAs, o grupo de desenvolvimento recebia também as seguintes informações relacionadas a cada tutorial a ser desenvolvido: 1) descrição, objetivos e conteúdos do OA; 2) novos elementos de programação apresentados; 3) nome do arquivo que será gerado; 4) nome do arquivo do programa desenvolvido; e 5) descrição do programa em C, contendo código inicial e final do mesmo, e valores de entrada sugeridos (quando fosse o caso). Após desenvolvidos, os tutoriais eram disponibilizados na intranet do laboratório de informática e na internet.

4.3. Conteúdos Abordados

Ao todo foram desenvolvidos 16 (dezesseis) OAs que abordam alguns dos conteúdos básicos do currículo da disciplina de AP. A Tabela 1 apresenta os tutoriais desenvolvidos, seus objetivos e os conteúdos abordados:

Tabela 1. Objetos de Aprendizagem Desenvolvidos

N	Título - Objetivo	Conteúdos
01	Uso do Editor Kate (Para usar no Linux) - Introduzir o editor Kate como uma ferramenta para a digitação e compilação de códigos em C.	1) Formas de abrir o editor Kate no Linux; 2) Como criar e salvar um arquivo; 3) As partes do editor Kate; 4) Como acessar o terminal console no Kate.

02	Olá Mundo (usando o Kate e o gcc no Linux) - Desenvolver, compilar e executar um programa 'Olá Mundo' em C	1) Delimitadores de bloco; 2) Biblioteca Stdio.h; 3) Estrutura de um programa em C; 4) Significado do main(void) e do return; 5) Como escrever mensagens na tela. Como pular de linha; 6) Como e onde compilar, (gcc -o saída teste.c) explicar que tem q ser feito no console; 7) Como e onde executar; 8) Onde os resultados são apresentados;
03	Olá Mundo (usando a IDE Dev C++ no Windows) - Desenvolver, compilar e executar um programa Olá Mundo em C.	1) Como criar e salvar um arquivo; 2) Delimitadores de bloco; 3) Biblioteca Stdio.h; 4) Estrutura de um programa em C; 5) Significado do main(void) e do return; 6) Como escrever mensagens na tela. Como pular de linha; 7) Como e onde compilar; 8) Como e onde executar; 9) Onde os resultados são apresentados; 10) Uso do comando system("PAUSE") para visualizar os resultados; 11) Web site onde se baixa a ferramenta;
04	Leitura e Escrita - Apresentar os tipos de variáveis int, float e char e como são realizadas suas declarações e introduzir o uso do comando de leitura scanf().	1) Declaração de variáveis inteiras, reais e caracteres; 2) Solicitação de valores de entrada para o usuário; 3) Uso do comando de leitura scanf(); 4) Formatação de leitura de tipos de variáveis no C (%d %c %i).
05	Cálculo de Média - Apresentar o comando de atribuição "=" e o uso dos operadores "+" e "/", e dos parênteses "(", ")".	1) Comando de atribuição; 2) Realização de operações aritméticas com variáveis reais "+", "/"; 3) Utilização de Parênteses em operações aritméticas.
06	Utilização de Parênteses em Operações Aritméticas - Introduzir a utilização de parênteses como fundamentais na ordem de execução de operações aritméticas.	1) Erro no cálculo da média entre dois valores reais por falta de utilização de parênteses.
07	Operadores Aritméticos e Relacionais - Apresentar os operadores aritméticos de divisão inteira e resto de divisão inteira, apresentar os operadores relacionais > e <, e demonstrar que resultados de operações relacionais em C são do tipo inteiro, sendo 0 para verdadeiro e 1 para falso.	1) Operadores aritméticos para divisão inteira e resto de divisão inteira; 2) Operadores relacionais > e <; 3) Valores resultantes de operações relacionais.
08	Estrutura de Condição IF ELSE - Introduzir o uso da estrutura de condição IF ELSE e apontar a necessidade de estruturas de condição aninhadas.	1) Estrutura de condição IF ELSE; 2) Avaliação de expressões na estrutura de condição IF ELSE; 3) Escopo da estrutura IF ELSE (uso das chaves); 4) Demonstração de qual cláusula (if ou else) é executada dependendo dos valores de entrada.
09	Operador Lógico E - Apresentar o operador Lógico de Conjunção (E).	1) Uso do operador E para definir intervalos entre números.
10	Operador Lógico OU - Apresentar o operador Lógico de Disjunção (OU) e apontar a necessidade de estruturas de condição aninhadas.	1) Uso do operador lógico OU para avaliar se o valor de uma variável sexo do tipo caractere foi digitado corretamente.
11	Estrutura de Condição IF ELSE Aninhada - Apontar a necessidade de uso da estrutura de condição IF ELSE	1) Estrutura de condição IF ELSE aninhada; 2) Escopo da estrutura IF ELSE aninhada (uso das chaves); 3) Demonstração de qual cláusula (if ou

	aninhada.	else) é executada dependendo dos valores de entrada.
12	Estrutura de Condição Switch - Demonstrar o funcionamento da estrutura de condição Switch no desenvolvimento de um pequeno menu de opções.	1) Estrutura de condição Switch; 2) Escopo da estrutura Switch; 3) Restrições de uso da estrutura de condição Switch (somente variáveis ordinais).
13	Estrutura de Repetição For (com contagem crescente) - Apresentar a estrutura de repetição For.	1) Estrutura de repetição For; 2) Uso de contadores; 3) Incremento de um contador i pertencente ao laço For.
14	Estrutura de Repetição For (com contagem decrescente) - Apresentar a estrutura de repetição For com contagem decrescente.	1) Estrutura de repetição For; 2) Uso de contadores; 3) Incremento de um contador i pertencente ao laço For.
15	Estrutura de Repetição For (com contagem crescente e alfanumérica) - Demonstrar a utilização da estrutura de repetição For com contagem alfanumérica crescente.	1) Estrutura de repetição For; 2) Uso de contadores; 3) Incremento de um contador alfanumérico pertencente ao laço For.
16	Estrutura de Repetição WHILE - Demonstrar a utilização da estrutura de repetição While e comparar a estrutura de repetição While com a estrutura de repetição For.	1) Estrutura de repetição While; 2) Contadores; 3) Inicialização de contadores.

A figura 1 apresenta duas telas do tutorial “Olá Mundo (usando o Kate e o gcc no Linux”, em que são abordadas informações relacionadas ao escopo de um programa C e como o programa pode ser compilado utilizando o gcc.

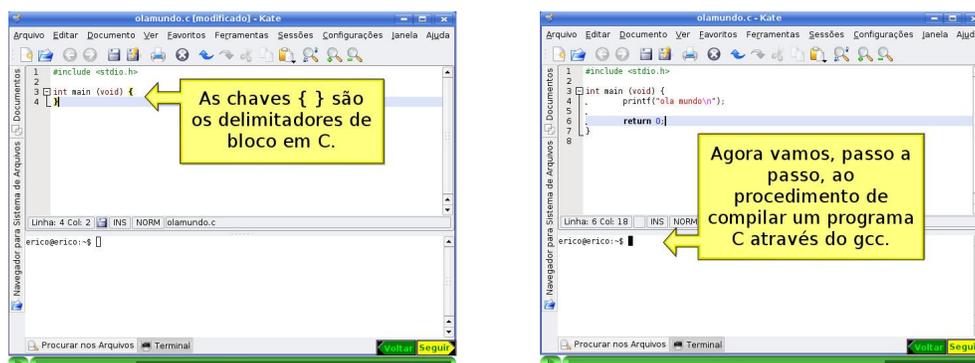


Figura 1. Telas do Tutorial “Olá Mundo (usando o Kate e o gcc no Linux)”

5. Avaliação junto aos Acadêmicos

Para avaliar a utilização e a aceitação dos OAs por parte dos acadêmicos foi realizado um levantamento de dados junto aos mesmos. O formato do questionário aplicado e a metodologia foram extraídos e adaptados de [Lahtinen *et al.* 2005]. O questionário foi subdividido em 4 (quatro) seções: experiências prévias do acadêmico, dificuldades enfrentadas no aprendizado, materiais de apoio ao processo de ensino-aprendizagem, e opiniões sobre os tutoriais disponibilizados.

Na seção 1 foram abordadas questões relacionadas ao tempo em que cada aluno tinha de experiência em programação, como o mesmo julgava suas habilidades antes e depois de cursar a disciplina, e a quantidade de tempo semanalmente dedicada para estudar e praticar os conteúdos abordados.

A seção 2 buscou identificar as principais dificuldades enfrentadas pelos acadêmicos no processo de ensino-aprendizagem da disciplina. As respostas para as questões pertenciam a uma escala de 5 pontos, *muito fácil de aprender* (1) a *muito difícil de aprender* (5). Para as perguntas dessa seção e das seções seguintes foi disponibilizada também uma opção *não sei*.

O objetivo da seção 3 foi levantar quais os principais tipos de materiais ajudaram ou ajudariam o acadêmico a aprender programação. As respostas pertenciam a uma escala de 5 pontos, *praticamente inútil* (1) a *muito útil* (5).

A última seção tinha como objetivo avaliar os OAs desenvolvidos e indicar os interesses dos acadêmicos no desenvolvimento de novos OAs. As opiniões sobre os objetos desenvolvidos foram recolhidas em uma escala de 1 a 5, *praticamente inútil* (1) a *muito útil* (5). As perguntas sobre o interesse dos acadêmicos em novos tutoriais ofereciam uma escala de 1 a 4 pontos, *nenhum interesse* (1) a *alto interesse* (4).

6. Resultados obtidos

O questionário foi aplicado com duas turmas de AP que continham acadêmicos de primeira fase do curso de Engenharia de Computação e repetentes desse e de outros cursos da instituição. O semestre iniciou com 76 acadêmicos, porém somente 34 responderam ao questionário (tendo em vista as desistências ao longo do semestre). A tabela 2 apresenta um resumo das informações levantadas junto aos acadêmicos.

6.1 Experiências prévias do acadêmico, dedicação e opiniões sobre o aprendizado

Os dados mostraram que 67% dos acadêmicos tiveram seu primeiro contato com programação nessa disciplina. Os acadêmicos manifestaram sua opinião quanto ao aprendizado de AP da seguinte maneira: 47% consideraram o aprendizado prazeroso, 35% desafiante, 12% difícil, 3% entediante e 3% igual a de outras disciplinas. Ainda, de acordo com o levantamento, os acadêmicos dedicaram para seus estudos uma média de 3,5 horas por semana, além do tempo estudo em sala de aula.

6.2 Dificuldades Enfrentadas no Aprendizado

Os acadêmicos responderam que a situação em que sentem maior dificuldade no aprendizado de programação é a de *encontrar erros no próprio programa (erros de compilação)* (I7), seguido de *desenvolver um programa para executar uma determinada tarefa* (I5) e *dividir funcionalidades em procedimentos/funções*(I6). Os conteúdos considerados mais difíceis foram *Funções e Procedimentos* (C10), seguidos por *Manipulação de erros* (C13) e *Arrays (vetores)* (C9).

6.3 Sobre o Processo de Ensino-Aprendizagem

Questionados sobre os tipos de materiais que mais ajudariam/ajudaram no processo de ensino-aprendizagem da disciplina, as *visualizações interativas* (M7)

ficaram em quarto lugar, atrás de *exemplos de programas* (M4), *exemplos de programas com problemas* (M5) e *notas de aula/cópias de transparências* (M2).

Tabela 2. Resumo dos Dados Recolhidos

Pergunta	Cod.	Num. Alunos	Média
SOBRE OS CONTEÚDOS DO CURSO			
Em que tipo de situação você sente dificuldade no aprendizado de programação?			
Uso do ambiente de desenvolvimento de programação	I1	33	1,82
Uso do Compilador ou IDE (Dev-C++, gcc)	I2	33	1,97
Entender as estruturas de programação	I3	34	2,56
Aprender a sintaxe da linguagem de programação	I4	34	2,76
Desenvolver um programa para executar uma determinada tarefa	I5	33	3,09
Dividir funcionalidades em procedimentos/funções	I6	34	3,06
Encontrar erros no próprio programa (erros de compilação)	I7	34	3,24
Em que tipo de conteúdo você sente dificuldade no aprendizado de programação?			
Operadores (lógicos, relacionais e aritméticos) e sua hierarquia	C1	34	1,68
Entrada e Saída de Dados	C2	34	1,44
Variáveis	C3	34	1,53
Estruturas de Seleção	C4	34	2,09
Estruturas de Seleção aninhadas	C5	34	2,15
Estruturas de repetição	C6	34	2,12
Estruturas de repetição aninhadas	C7	34	2,38
Recursividade	C8	26	2,92
Arrays (vetores)	C9	34	2,97
Funções e Procedimentos	C10	34	3,21
Tipos de dados estruturados	C11	29	2,79
Manipulação de entrada e saída de dados	C12	32	2,69
Manipulação de erros	C13	33	3,12
Uso de bibliotecas das linguagens	C14	32	2,59
SOBRE OS MATERIAIS DE APOIO AO PROCESSO DE ENSINO APRENDIZAGEM			
Que tipos de materiais ajudaram (ou ajudariam) você a aprender programação?			
Livro texto de programação	M1	33	3,88
Notas de aula/cópias de transparências	M2	33	4,21
Exercícios de pergunta e resposta	M3	34	3,76
Exemplos de programas	M4	34	4,71
Exemplos de programas com problemas	M5	34	4,21
Figuras estáticas de estruturas de programas	M6	31	3,61
Visualizações interativas (vídeos de tutorial)	M7	34	4,15
SOBRE OS TUTORIAIS DISPONIBILIZADOS			
Qual sua avaliação para cada tutorial disponibilizado?			
Uso do Editor Kate	T1	34	4,50
Olá Mundo (usando gcc no Linux Kurumin)	T2	34	3,88
Olá Mundo (usando a IDE Dev C++ no Windows)	T3	34	3,85
Leitura e Escrita	T4	34	4,24
Utilização de Parênteses em Operações Aritméticas	T5	34	4,26
Cálculo de Média entre dois números inteiros	T6	34	4,09
Operadores Aritméticos e Relacionais	T7	34	4,12
Estrutura de Condição IF ELSE	T8	34	4,24
Operador Lógico E	T9	34	3,97
Operador Lógico OU	T10	34	3,82
Estrutura de Condição IF ELSE Aninhada	T11	34	3,94
Estrutura de Condição SWITCH	T12	34	4,06
Estrutura de Repetição FOR (contagem crescente)	T13	33	4,03
Estrutura de Repetição FOR (contagem decrescente)	T14	33	4,00
Estrutura de Repetição FOR (contagem crescente e alfanumérica)	T15	33	3,76

Estrutura de Repetição WHILE	T16	31	4,13
------------------------------	-----	----	------

Quantifique seu interesse por novos tutoriais envolvendo os seguintes assuntos			
Vetores	Z1	32	3,28
Matrizes	Z2	32	3,59
Funções e Procedimentos	Z3	32	3,75
Passagem de Parâmetros por valor e por referência	Z4	32	3,81
Ponteiros	Z5	32	3,69
Estrutura de dados	Z6	32	3,34
Arquivos	Z7	32	3,28
Erros de Compilação em C	Z8	32	3,38
Erros de Sintaxe em C	Z9	32	3,09

6.4 Sobre os tutoriais disponibilizados e seu uso

No que se refere ao local onde os mesmos costumavam utilizar os OAs, 70% responderam que utilizavam em casa ou no trabalho, 18% que usavam no laboratório de informática da universidade em horário de aula, 3% que usavam no laboratório de informática fora do horário de aula, e 9% responderam que nunca usaram os tutoriais. Os OAs considerados mais úteis pelos acadêmicos foram: *Uso do Editor Kate* (T1), *Operadores Aritméticos e Relacionais* (T7), *Leitura e Escrita* (T4), e *Estrutura de Condição IF ELSE* (T8).

Quando questionados quanto ao interesse em relação a criação de novos tutoriais que abordassem assuntos ainda não contemplados nos OAs já desenvolvidos, 94% dos entrevistados responderam que tinham interesse e apenas 6% responderam que não tinham interesse. Os assuntos que foram apontados como mais interessantes para serem abordados em novos tutoriais são: *Passagem de Parâmetros por valor e por referência* (Z4), *Funções e Procedimentos* (Z3), *Ponteiros* (Z5) e *Matrizes* (Z2).

7. Considerações Finais

Considera-se que o impacto causado pelos objetos de aprendizagem foi positivo. Através da pesquisa constatou-se que o primeiro tutorial disponibilizado foi o mais requisitado e acessado. O professor da disciplina percebeu maior facilidade em conduzir as aulas com o auxílio dos tutoriais e observou um melhor rendimento por parte dos acadêmicos, pois os mesmos tinham acesso a um material interativo para consulta durante as aulas. Percebeu-se inclusive, que em aulas iniciais da disciplina foi possível executar juntamente com os alunos uma quantidade maior de exercícios do que quando não havia o auxílio dos tutoriais.

Entende-se que a utilização dos tutoriais fora da faculdade por parte dos alunos influenciou no rendimento das aulas, e incentivou nos mesmos o desenvolvimento de uma capacidade autodidata e de independência na resolução de problemas.

Em uma contagem do número de acessos ao site onde os OAs foram disponibilizados, foi constatada uma diminuição progressiva na utilização dos tutoriais a medida em que a disciplina transcorria. Conclui-se que após adquirir experiência com a programação, o usuário dos tutoriais não sente tanta necessidade de buscar ajuda em vídeos, e sim, em objetos mais abrangentes e aprofundados.

Como trabalhos futuros, espera-se desenvolver novos OAs para conteúdos ainda não abordados tomando como ponto de partida as informações já recolhidas junto aos

acadêmicos, e avaliar de forma mais direta o impacto do uso desses materiais no desempenho das disciplinas de AP.

8. Referências

- Brennan, M.; Funke, S.; Anderson, C. (2001). The Learning Content Management System: A New eLearning Market Segment Emerges. An IDC White Paper. Disponível em: <www.e-learning.com/download/indeedown.htm>. Acesso em: 05 fevereiro 2006.
- Dos Santos, R.P.; Costa, H.A.X. (2005). TBC-AED e TBC-AED/WEB: Um Desafio no Ensino de Algoritmos, Estruturas de Dados e Programação. In Anais do IV WEIMIG - Workshop em Educação em Computação e Informática do Estado de Minas Gerais. Varginha: Brasil.
- Downes S. (2001). Learning Objects: Resources for Distance Education Worldwide. *In* International Review of Research in Open and Distance Learning. Volume 2 (1).
- Falkembach, G. A. M.; Amoretti, M. S. M.; Tarouco, L. R., Viero, F. (2003). Aprendizagem de Algoritmos: Uso da Estratégia Ascendente de Resolução de Problemas. 8º Taller Internacional de Software Educativo. Santiago, Chile.
- Garcia, I. C.; De Rezende, P. J.; Calheiros, F. C. (1997). Astral: Um Ambiente para Ensino de Estruturas de Dados através de Animações de Algoritmos. In Revista Brasileira de Informática na Educação. Florianópolis: SC, Volume. 1, p. 71-80.
- L'Allier, J.J. (1997). Frame of Reference: NETg's Map to the Products, Their Structure and Core Beliefs. NetG. 1997. Disponível em: <www.im.com.tr/framerefer.htm#_TocFocalpoint>. Acesso em: 04 fevereiro 2006.
- Lahtinen, E.; Alamutka, K.; Järvinen, H. (2005) A Study of the Difficulties of Novice Programmers. ITiCSE'05, June 27–29, Monte de Caparica, Portugal
- Longmire, W. (2000). A Primer on Learning Objects. Learning Circuits on-line. 2000. Disponível em <www.learningcircuits.org/2000/mar2000/Longmire.htm>. Acesso em: 04 fevereiro 2006.
- Olsen, A.L. (2005). Using Pseudocode to Teach Problem Solving. *In* Journal of Computing Sciences in Colleges, Volume 21, Issue 2, December. Consortium for Computing Sciences in Colleges, USA.
- Raabe, A.L.A.; Silva, J.M.C. (2005). Um Ambiente para Atendimento as Dificuldades de Aprendizagem de Algoritmos. *In* Anais do XXV Congresso da Sociedade Brasileira de Computação: XIII Workshop sobre Educação em Computação. São Leopoldo: Brasil, p. 2326-2337.
- Smith, Jeffrey W. (1981). A Method for Teaching Programming. *In* Proceedings of the twelfth SIGCSE technical symposium on Computer science education: Technical Symposium on Computer Science Education. St. Louis, Missouri: United States, p. 252-255. ACM Press New York, NY, USA.