

# Um Ambiente Interativo de Aprendizagem para o ensino de programação na Universidade de Brasília

Homero L. Piccolo<sup>1</sup>, Marcus O. da Silva<sup>1</sup>, Vinícius A. Rodrigues<sup>1</sup>, Yuri A. N. Maia<sup>1</sup>

<sup>1</sup>Departamento de Ciência da Computação – Universidade de Brasília  
Caixa Postal 4466 – 70910-970 - Brasília - DF - Brasil

[homero@unb.br](mailto:homero@unb.br), [marcus\\_oliveiral5@hotmail.com](mailto:marcus_oliveiral5@hotmail.com), [vinicius.a.ro@gmail.com](mailto:vinicius.a.ro@gmail.com),  
[maiayuri@hotmail.com](mailto:maiayuri@hotmail.com)

**Abstract.** *This article describes the ICC Tutorial, developed at the Department of Computer Science at the University of Brasilia to be used for teaching the course Introduction to Computer Science (ICC), which goes through a transformation to a b-learning course. The ICC course is offered to a large number of students from different areas, and most of its content consists on learning the basic mechanisms of programming.*

**Resumo.** *Este artigo descreve o Tutorial ICC, desenvolvido no Departamento de Ciência da Computação da Universidade de Brasília para ser utilizado no ensino da disciplina Introdução à Ciência da Computação (ICC), que passa por um processo de transformação para a forma semi-presencial. A disciplina ICC é oferecida a um grande número de estudantes de diversas áreas, e a maior parte do conteúdo consiste na aprendizagem dos mecanismos básicos de programação.*

## 1. Introdução

O Tutorial ICC é um ambiente interativo no processo de ensino/aprendizagem da disciplina Introdução à Ciência da Computação (ICC), desenvolvido no Departamento de Ciência da Computação da Universidade de Brasília. Esta disciplina é ofertada em 10 turmas a mais de 300 alunos, e utiliza a linguagem Pascal para o ensino de programação. As turmas estão sendo gradativamente transformadas para a modalidade semi-presencial, na qual não há aulas presenciais, mas os alunos devem comparecer para realizar as provas na Universidade. Atualmente 3 turmas têm suas aulas nesta modalidade utilizando o Tutorial ICC.

O conteúdo do curso de ICC está bem consolidado, não se altera de um semestre para outro e, além disso, o número de alunos é grande. Esses fatores justificam a opção por um curso semi-presencial, desde que se disponha de um bom material didático. O desafio é a construção de um ambiente computacional que considere a característica do curso, que é desenvolver no aluno competência em programação de computadores. É preciso que o aluno entenda certos mecanismos e procedimentos, e os utilize para escrever programas que executem corretamente no computador, solucionando problemas de suas respectivas áreas.

## 2. Referencial teórico

Um dos pioneiros em Sistemas Tutores Inteligentes (STI), John Self, no artigo [Self 1999] elabora um perfil simples e claro do que deve ser um STI: “*sistemas de aprendizagem apoiados em computador, que tentam se adaptar às necessidades dos aprendizes e são, portanto os únicos sistemas que tentam ‘interessar-se’ pelos aprendizes*”. O Tutorial ICC, sem pretender ainda ser um STI, inicia seu caminho nessa direção com a possibilidade de adaptar o conteúdo à capacidade e necessidade de cada aluno. O estudo dos parâmetros para conferir essa capacidade de adaptação ao Tutorial ICC deu origem a uma monografia de conclusão de curso de Graduação [Brandizzi 2009].

Pode-se dividir, de forma simplificada, os conhecimentos em dois tipos: declarativos e procedurais. De acordo com [Fink 1991], os conhecimentos declarativos referem-se a fatos, figuras e conceitos que devem ser expostos ao aluno. Conhecimentos procedurais podem ser entendidos como regras a seguir para resolver um problema, como por exemplo, a resolução de uma questão de cálculo, ou a elaboração de um programa de computador. Os conhecimentos de tipo procedural são os que melhor uso podem fazer de um STI. A capacidade de adaptação às necessidades de cada aluno, que o Tutorial ICC oferece nos exemplos, vem ao encontro da necessidade do aluno adquirir uma habilidade no curso de ICC, e não apenas aprender um conjunto de conceitos teóricos.

## 3. Recursos do Tutorial ICC

O Tutorial ICC tem vários recursos interessantes, dos quais serão abordados aqui três: Simulador Interativo, Adaptabilidade e Correção Automatizada. Todos foram desenvolvidos integralmente neste projeto, utilizando linguagem Java.

### 3.1. Simulador Interativo (Tira-teima)

Uma das principais ferramentas do Tutorial ICC é o simulador interativo que denominamos Tira-teima. Consiste em um simulador da execução de programas, com visualização gráfica do estado da memória, dos arquivos e do console de um computador, ao longo da execução de um programa.

O Tira-teima tem a aparência mostrada na Figura 1. Na parte esquerda aparece o código do programa, em Pascal, que está sendo estudado. O aluno pode percorrer o código linha a linha. As linhas são numeradas para facilitar o aluno a se localizar, e uma faixa em *high lighth* mostra qual o comando que está sendo simulado no momento.

O lado direito da tela é dividido em quatro partes: a primeira mostra cada uma das variáveis que estão sendo utilizadas pelo programa. Na Figura 1 há uma matriz de 4 x 4 inteiros, além de outras variáveis simples.

A segunda parte da tela, do lado direito, mostra os arquivos que estão abertos no momento. Na Figura 1 há apenas um arquivo aberto, contendo 16 números inteiros.

A terceira parte da tela, do lado direito, faz uma simulação do console enquanto o programa é executado. Nessa janela preta aparecem tanto as mensagens que o programa escreve na tela como os eventuais dados que o usuário fornece ao programa.

A quarta parte da tela é uma área reservada para explicações que o professor queira dar ao aluno em determinadas linhas de código. Essa janela, em geral, fica fecha-

da. Ela só é aberta nos pontos em que o professor julga necessário fazer um comentário. Ela tem fundo vermelho, para chamar a atenção do aluno, e sua abertura pode ser acompanhada por um bip sonoro.

O Tira-teima permite a representação visual de qualquer estrutura de dados estática: variáveis simples como *integer*, *real*, *boolean*, *string* e *char*, ou compostas, como *arrays* e *records*, com todas as suas combinações possíveis: um *array* dentro do *record*, um *array* de *records*, etc.

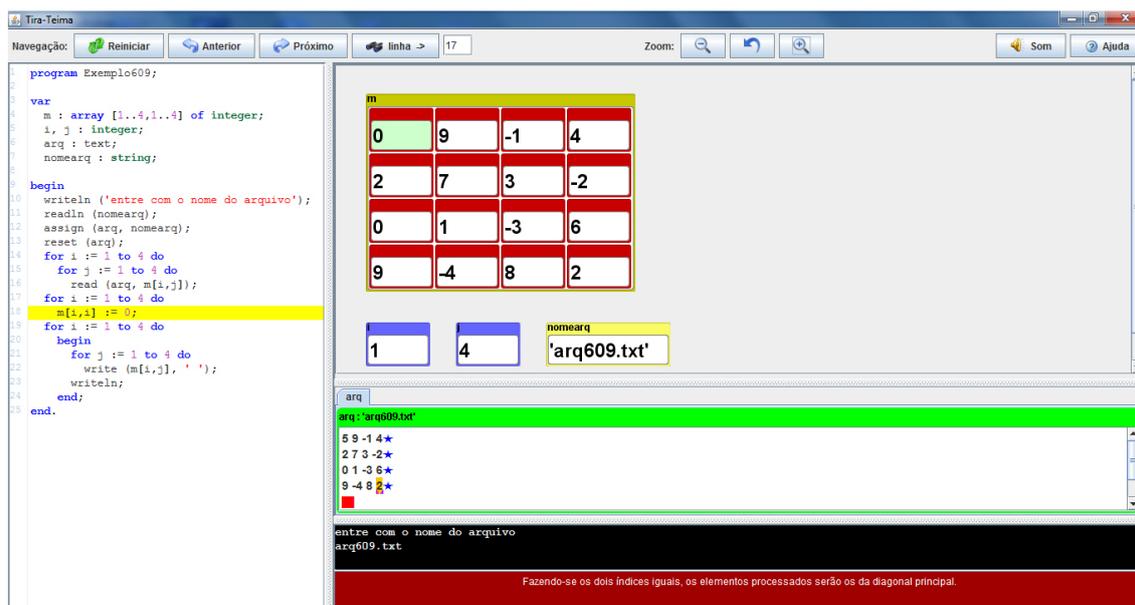


Figura 1. Interface do Tira-teima

### 3.2. Adaptabilidade

Outro recurso do Tutorial ICC é a capacidade de adaptar a sequência de apresentação do conteúdo ao nível e necessidade de cada aluno.

Em cada capítulo há um item denominado Exemplos de Aplicação. Este tópico disponibiliza grande parte da informação dada ao aluno, na forma de exemplos de programas. Nessa parte do Tutorial foi embutida certa inteligência na distribuição do conteúdo, de forma a permitir vários caminhos alternativos para o aluno. O Tutorial, de certa forma, é capaz de se adaptar à necessidade de cada aluno.

Os exemplos de aplicação são organizados em forma arborescente, de modo que na raiz da árvore fique o exemplo mais complexo e nas folhas os mais simples. Foi desenvolvida uma taxonomia de programas. Inicialmente elaborou-se uma lista de tópicos ou conceitos que devem ser transmitidos aos alunos. Esses conceitos foram reunidos em grupos de dois. Em cada grupo, estabeleceu-se uma hierarquia arborescente, de tal forma que o programa da raiz contém todos os tópicos, e os programas das folhas contém apenas uma característica cada um, ou eventualmente nenhuma.

Nos exemplos situados nos níveis mais baixos das árvores, há uma maior quantidade de mensagens enviadas pelo Tira-teima correspondente, para facilitar a compreensão por parte do aluno.

### **3.3. Correção Automatizada**

Os alunos de ICC no curso semi-presencial são avaliados por meio de provas realizadas no laboratório de informática da UnB. As provas consistem em programas em Pascal que o aluno deve elaborar, podendo compilar e executar quantas vezes quiser no tempo pré-determinado para cada prova. No final da prova o aluno envia os programas pelo Moodle ao professor.

O professor corrige as provas por meio do Corretor Automático, que executa os programas de cada aluno com as entradas de dados previstas pelo professor, e verifica se as saídas de dados do programa do aluno conferem com as saídas previstas pelo professor.

As questões são elaboradas de modo que o Módulo de Avaliação possa efetuar vários testes para cada programa. Dessa forma, podem ser testadas condições parciais, e a nota não fica restrita a 0 ou 10, mas procura-se que haja um espalhamento das notas entre 0 e 10.

O Corretor Automático emite um relatório para cada aluno, com as notas globais e as notas de cada teste efetuado. A prova do aluno fica armazenada e o aluno, se quiser, pode fazer a revisão de sua prova com os monitores da disciplina. Esta sistemática agiliza a correção de provas de um número elevado de alunos sem perda de qualidade, pois as provas consistem em programas que o aluno deve construir.

## **4. Conclusão**

O Tutorial ICC está sendo testado gradualmente. No semestre atual, três das dez turmas o estão utilizando. Uma avaliação rigorosa deve ser efetuada mais adiante, mas os resultados provisórios estão sendo positivos: os alunos preferem utilizar o sistema de curso semi-presencial em lugar do presencial, e a qualidade do ensino de ICC está sendo mantida.

## **5. Referências**

- Brandizzi, A. V. N., Nuñez, (2009) “Taxonomia de Problemas para Aprendizagem de Computação”, Monografia de Graduação do Curso de Licenciatura em Ciência da Computação, Universidade de Brasília.
- Fink, P. (1991) “The Role of Domain Knowledge in the Design of an Intelligent Tutoring System”, In: Burns, H.; Parlett, J.W.; Redfield, C.L. “Intelligent Tutoring Systems - Evolutions in Design”, Mahwah, New Jersey, Lawrence Erlbaum Associates Publishers.
- Self, J. (1999) “The defining characteristics of intelligent tutoring systems research: ITSs care, precisely”. In: International Journal of Artificial Intelligence in Education, v. 10, p. 350-364.