TIRA-TEIMA: UMA FERRAMENTA PARA APRENDIZAGEM DE PROGAMAÇÃO A DISTÂNCIA

Homero Luiz Píccolo[1]

Abstract — Tira-teima is a program developed at the Department of Computer Science at the University of Brasilia in order to support programming courses in a distance learning environment. It consists of an application built in Java, to simulate the execution of a software written in Pascal. Tira-teima enables students to view, for a program, step by step, the variables declared and their current values. It also allows the student to view the files in use and which element is being read or written to the console. It has a special area to show to students some explanation that the teacher considers necessary in a particular step. This tool can be seen within the Tutorial ICC, used for teaching Introduction to Computer Science at UnB (www.cic.unb.br/~homero/tutorial icc).

Index Terms – Introduction to Computer Science, Visualization Tool, Distance Learning.

INTRODUÇÃO

O Tira-teima é um programa desenvolvido no Departamento de Ciência da Computação da Universidade de Brasília com a finalidade de apoiar cursos de programação na modalidade a distância. Consiste em um simulador de execução de programas elaborados em linguagem Pascal.

Dado um programa, o Tira-teima permite que o aluno visualize, para cada passo, as variáveis declaradas e seus valores atuais, os arquivos em uso e qual elemento está sendo lido ou gravado, o console com as mensagens do programa e do usuário, e uma área especial para mostrar aos alunos alguma explicação que o professor julgue necessária em um determinado passo.

O professor pode escolher o tamanho, posição e cores das representações gráficas. Foi construído em Java. Os recursos de orientação a objetos facilitam a criação e visualização 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.

Ao percorrer um programa, o aluno pode andar para frente ou para trás no programa, refazendo qualquer trecho quantas vezes quiser. Essa ferramenta pode ser vista dentro do Tutorial ICC, utilizado para o ensino de Introdução à Ciência da Computação na UnB (www.cic.unb.br/~homero/tutorial icc).

CONTEXTO

O Tira-teima é a principal ferramenta do Tutorial ICC, que serve de suporte à disciplina Introdução à Ciência da Computação na modalidade semi-presencial. Os alunos devem aprender a programar em linguagem Pascal sem assistir aulas. As provas consistem em programas que os alunos dever construir individualmente nos laboratórios da Universidade. O Tutorial ICC está sendo construído para ser um Sistema Tutor Inteligente (STI). Um dos pioneiros em Sistemas Tutores Inteligentes (STI), John Self [5] 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 ser ainda um STI pleno, 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 [2].

Pode-se dividir, de forma simplificada, os conhecimentos em dois tipos: declarativos e procedurais. De acordo com Fink [3], os conhecimentos declarativos referem-se a fatos, figuras, 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 vem ao encontro da necessidade do aluno de adquirir uma habilidade no curso de ICC, e não apenas de aprender um conjunto de conceitos teóricos.

Segundo Nielsen [4] a Usabilidade é um atributo de qualidade relacionado à facilidade do uso de algo. Mais especificamente, refere-se à rapidez com que os usuários podem aprender a usar alguma coisa, a eficiência deles ao usá-la, o quanto lembram daquilo, seu grau de propensão a erros e o quanto gostam de utilizá-la. Na construção do Tira-teima procurou-se seguir as indicações dos princípios de Usabilidade, daí resultando uma monografia de conclusão de curso de Graduação [1].

1 Homero Luiz Píccolo, Professor Adjunto do Departamento de Ciência da Computação da Universidade de Brasília, homero@unb.br

DESCRIÇÃO DO TIRA-TEIMA

O Tira-teima tem a aparência mostrada na Figura 1. Na parte esquerda da tela aparece o código do programa 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 ligth* mostra qual o comando que está sendo simulado no momento.

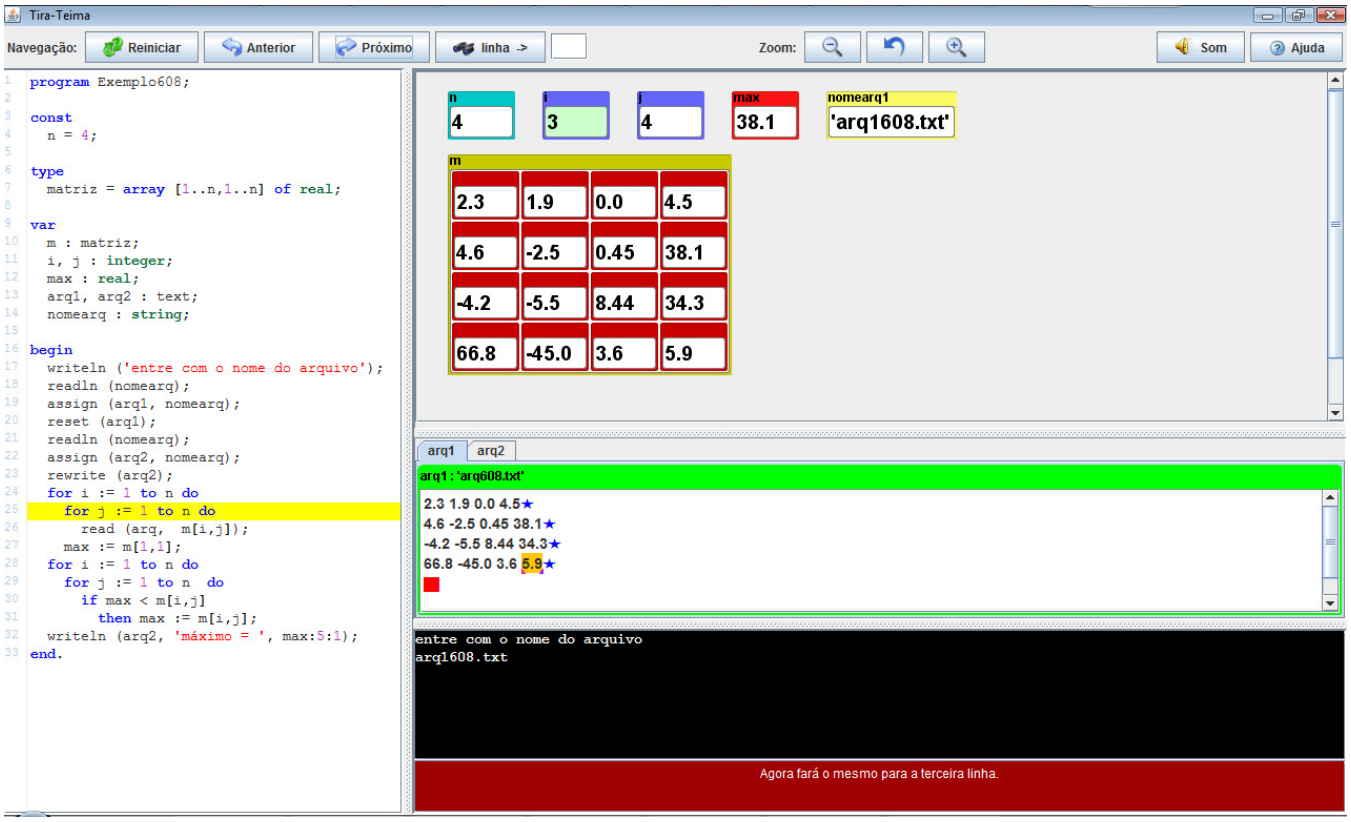

FIGURA 1
APARÊNCIA DO TIRA-TEIMA

FIGURA 2
REPRESENTAÇÃO DE UM ARRAY DE RECORDS

O lado direito da tela é dividido em quatro partes: a primeira mostra cada uma das variáveis que estão sendo utilizadas pelo programa. Em cada variável, na parte colorida externa aparece o nome da variável e na janela branca interna aparece o seu valor. Na Figura 1, as variáveis n, i, e j são inteiras e valem respectivamente 4, 1 e 4. A variável max é real e vale 2.3. A variável nomearq é uma cadeia de caracteres e seu valor é "arq608.txt". A variável m é uma matriz 4x4 de inteiros.

A segunda parte da tela, do lado direito, mostra os arquivos que estão abertos no momento. Na janela aparece o conteúdo de um arquivo por vez. Na Figura 1, há dois arquivos abertos: arq1, cujo nome externo é "arq608.txt" e arq2, cujo nome externo é "arq609.txt". A janela mostra o conteú-

do do primeiro arquivo. O Tira-teima mostra de maneira automática o conteúdo do arquivo que está sendo utilizado no momento. No entanto, mesmo sem percorrer o programa, o aluno pode inspecionar o conteúdo dos outros arquivos, clicando na aba correspondente do arquivo desejado. No caso, o conteúdo do arquivo que aparece é uma matriz de 4 linhas contendo 4 números reais cada uma. Os arquivos utilizados são de tipo *text*; as estrelas azuis indicam o final de cada linha e o quadrado vermelho indica o final do arquivo. A marca alaranjada sobre um dos números acompanha o conteúdo do arquivo à medida que os dados são lidos.

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. No caso da Figura 1, o programa escreveu "entre com o nome do arquivo" e o usuário escreveu "arq608.txt".

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 fechada. 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. No caso da Figura 1 essa janela está aberta e a mensagem do professor ao aluno é: "Agora fará o mesmo para a terceira linha".

O Tira-teima funciona com programas previamente escolhidos pelo professor que prepara o Tutorial. Para cada programa deve ser escrito um roteiro, numa linguagem desenvolvida especificamente para o Tira-teima. Nessa linguagem, cada linha corresponde a um passo do programa (em um laço de repetição cada comando corresponde a várias linhas de roteiro). No Tutorial há 169 programas escolhidos como exemplos, com seus respectivos roteiros. Mas se um professor quiser criar novos exemplos, basta escrever seus próprios roteiros.

Ao percorrer um programa, o aluno pode andar para frente ou para trás no programa, refazendo qualquer trecho quantas vezes quiser. O Tira-teima permite também que o aluno pule um pedaço de código, para inspecionar um comando que lhe interesse. Esse recurso é particularmente útil para o caso de laços de repetição que o aluno não deseja visualizar passo a passo, mas prefere pular e verificar o resultado depois da execução do laço.

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. Na Figura 2 aparece a representação de um *array* de 5 elementos, denominado "alunos", no qual cada elemento é um *record* composto por uma string denominada "nome", um *integer* denominado "idade" e um *array* de 4 números reais, denominado "notas". A janela de visualização da memória permite aplicar *zoom* para inspecionar uma parte específica de uma estrutura qualquer. O professor que constrói o roteiro pode escolher a cor, tamanho e posição de todas as variáveis utilizadas pelo programa, criando assim uma aparência mais agradável e de acordo com padrões de usabilidade recomendados.

O aluno pode alterar o espaço destinado a cada uma das partes da tela (código, variáveis, arquivos, console e avisos), bastando colocar o mouse sobre alguma das linhas divisórias e arrastar da forma que desejar. Uma tecla de 'Ajuda' remete o aluno a instruções de uso da ferramenta.

UTILIZAÇÃO DO TIRA-TEIMA

O Tira-teima está sendo utilizado como uma ferramenta no Tutorial ICC, que é um tutorial construído também no Departamento de Ciência da Computação da Universidade de Brasília, para servir de suporte ao ensino da linguagem Pascal para alunos de vários cursos de Ciências Exatas, como as Engenharias, Matemática, Física, etc.

Isso se faz na disciplina Introdução à Ciência da Computação (ICC), que é ministrada a distância, na forma semi-presencial: os alunos comparecem apenas para fazer as provas.

Um dos recursos do Tutorial ICC é a capacidade de adaptar a sequência de apresentação do conteúdo ao nível e necessidade de cada aluno.

O Tutorial ICC tem, em cada capítulo, um item denominado Exemplos de Aplicação. Este tópico disponibiliza uma grande quantidade de informação para o aluno na forma de exemplos de programas que utilizam os mecanismos de programação ensinados.

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, que denominamos conceitos ou características do programa, que devem ser transmitidas aos alunos. Essas características foram reunidas em grupos de duas ou de três características. Em cada grupo, estabeleceu-se uma hierarquia, 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. Por exemplo, um programa que tem duas características, gera um conjunto como o da Figura 3.

No caso, o programa 1 tem as duas características, o programa 1.1 não tem nenhuma dessas duas características, o programa 1.2 tem uma das duas características e o programa 1.3 tem a outra característica.

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.

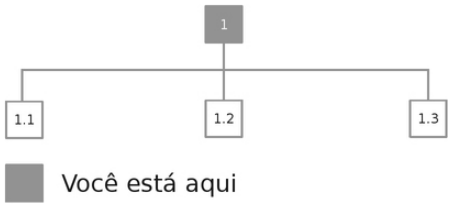

FIGURA 3
DISTRUBUIÇÃO DE PROGRAMAS COM DUAS CARACTERÍSTICAS

O Tutorial ICC apresenta ao aluno sempre o programa posicionado na raiz da árvore, e o sistema de navegação permite ao aluno escolher a trajetória que julgar mais conveniente. Dessa forma, o Tutorial se adapta à maior ou menor facilidade que o aluno tenha para compreender os exemplos.

Os alunos de ICC no curso semi-presencial são avaliados por meio de provas realizadas no laboratório da Universidade. 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 o programa pela Internet ao professor.

O professor corrige as provas por meio do Sistema de Correção Automática, que é um aplicativo Java desenvolvido em conjunto com o Tutorial ICC. O Sistema de Correção 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 Sistema de Correção 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 Sistema de Correção 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.

CONCLUSÃO

O Tira-teima tem se mostrado uma ferramenta muito útil na aprendizagem de programação em cursos a distância, como é o caso da disciplina Introdução à Ciência da Computação, do Departamento de Ciência da Computação da Universidade de Brasília. O Tira-teima está inserido no Tutorial ICC, desenvolvido para essa finalidade. Esse curso é oferecido na modalidade semi-presencial: não há aulas, mas as provas são feitas no laboratório da Universidade. Os alunos têm feito uma avaliação bastante positiva do Tira-teima, como a principal ferramenta do Tutorial ICC.

REFERÊNCIAS

[1] Biller, A. "Usabilidade na ferramenta Tira-teima do Tutorial ICC", *Monografia de Graduação do Curso de Licenciatura em Ciência da Computação*, Universidade de Brasília, 2009.

[2] Brandizzi, A., Nuñes Y. "Taxonomia de Exemplos para Aprendizagem de Computação", *Monografia de Graduação do Curso de Licenciatura em Ciência da Computação,* Universidade de Brasília, 2009

[3] Fink, P. "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",* Mahwad, New Jersey, Lawrence Erlbaum Associates Publishers, 1991.

[4] Nielsen, J., Loranger H. "Usabilidade na Web – Projetando Websites com Qualidade", Editora Campus, 2007.

[5] Self, J. "The defining characteristics of intelligent tutoring systems research: ITSs care, precisely". *In: International Journal of Artificial Intelligence in Education*, v. 10, p. 350-364, 1999.