**Sicas: Sistema interativo para desenvolvimento de algoritmos e simulação**

Características do programa:

**Para estudantes:**

* Inclui suporte de dois tipos de atividades:

1. design/edição de soluções para os problemas propostos.
   * + O estudante constrói algoritmos usando representações visuais, os fluxogramas.
2. execução/simulação dessas soluções.
   * + O estudante pode simular e animar a execução dos algoritmos que desenvolveu, analisando com detalhe e com o ritmo desejado dos vários passos e as entidades que participam.

Os estudantes podem selecionar um algoritmo feito por eles ou algoritmos do professor que tenham sido criados para esse problema.

**Para Professores:**

Para além dos dois tipos de atividades mencionados acima, o Sicas inclui ainda um modo professor que permite a criação e edição de uma biblioteca de problemas sugeridos. Conterá também uma solução definida pelo professor.

Pode abrir um algoritmo criado por um estudante.

Permite ao professor verificar se a solução do aluno apresentada resolve o problema.

A ideia da existência de vários algoritmos para o mesmo problema é permitir diferentes tipos de resolução e dar suporte a várias formas de compreensão, permitindo ao aluno comparar a sua resolução e estratégia com a solução mais adequada ao problema.

**Para ambos (Professor e Aluno):**

O utilizador pode consultar ou editar um dos seus algoritmos

O design do algoritmo é suportado por ícones onde o utilizador pode construir um fluxograma que representa as suas ideias ao invés de construir um algoritmo com pseudo-código, o que facilita a sua aprendizagem.

Pode-se tirar a conclusão de que:

* Demora menos tempo para compreender
* Produz-se menos erros por falta de compreensão
* Dá mais confiança aos alunos na compreensão do algoritmo
* Reduz o tempo de resposta relacionado com o algoritmo
* Reduz o número de vezes que os alunos precisam para analisar o algoritmo

As opções que os alunos podem usar quando criam fluxogramas no SICAS são:

* Atribuição – Define um valor de uma variável com o resultado de uma expressão
* Input/Output – Torna possível a leitura de valores do utilizador que escreve para a consola
* Repetição – Permite a repetição de execução de algumas acções. Temos como exemplo a figura 1 que mostra uma caixa de diálogo onde o estudante pode especificar as características da repetição.
* Seleção – Permite a escolha entre duas acções que poderão ser executadas

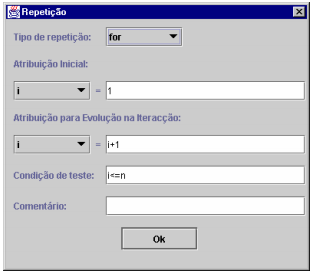


Fig.1 – Especificação de uma repetição

As caixas de diálogos usadas foram projectadas para incluir o mínimo de informação necessária para executar a estrutura do programa correspondente. Isto evita erros comuns de sintaxe nos programadores mais inexperientes e faz com que eles pensem mais cuidadosamente sobre os diferentes componentes de cada estrutura.

Os alunos podem facilmente eliminar, modificar ou copiar qualquer componente em qualquer altura. Pode ser feito através de menus, ícones correspondentes na barra de ferramentas ou activar/seleccionar menus com o botão direito do rato.

A aplicação SICAS suporta apenas o uso de quatro tipos de dados: “numeros”, “strings”, “arrays de strings” e “arrays de numeros”. O objectivo principal do SICAS é dar suporte á aprendizagem de programação básica. Considera-se que a diversidade de tipos de dados, apesar de ser um aspecto importante nas linguagens de programação, constitui, várias vezes, uma fonte de distração de dificuldade acrescentada nos alunos para o design do algoritmo.

A construção de expressões no SICAS usa uma sintaxe similar á linguagem C e Java. Essa decisão foi tomada em consideração aos alunos, que mais tarde irão programar numa dessas linguagens. No entanto tentaram minimizar os detalhes sintáticos, para que o aluno possa concentrar completamente na tarefa principal (criação do algoritmo que resolve o problema).

Outra característica importante é o suporte a funções definidas por alunos. Essa característica tenta introduzir os alunos aos conceitos de modularização. A modularização tem várias vantagens:

* Torna o algoritmo mais legível e fácil de compreender
* Ajuda a gerir e resolver problemas mais complexos e grandes
* Permite a alteração do módulo ou função sem alterar o resto do algoritmo
* Evita a redundância, tornando possível a redução do número de componentes

A construção de funções é feita usando as estruturas mencionadas acima de acordo com as regras e opções previamente descritas para qualquer resolução. O SICAS também tem funções pré definidas que podem ser utilizadas para a construção de expressões sem necessitar da sua declaração.

Existem funções para a manipulação de números e strings.

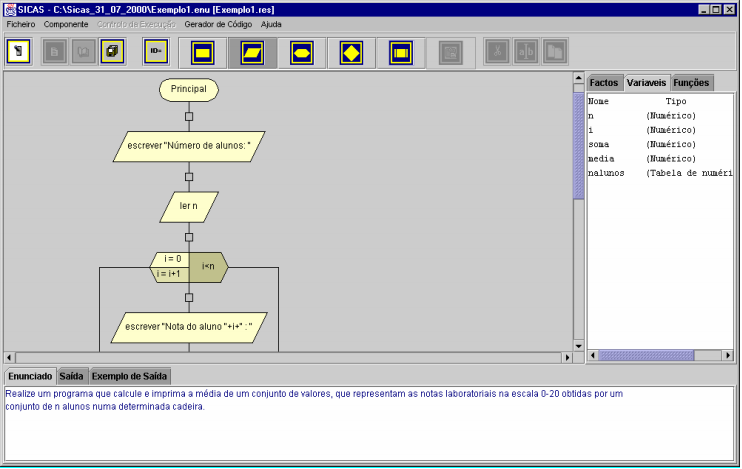


Figura 2 – Exemplo do desenho do algoritmo

Qualquer algoritmo criado com SICAS pode ser automaticamente traduzido para pseudo código, C ou JAVA. Estas alternativas não estão relacionadas com a sua importância, mas são a maneira de mostrar aos estudantes que um algoritmo bem desenhado pode ser facilmente traduzido em várias linguagens. Logo o algoritmo bem desenhado é o mais importante e não a linguagem que irá ser implementada para corresponder aos detalhes da sintaxe.

Depois de construído o algoritmo para solucionar um problema, o estudante/aluno pode ver uma simulação animada. O aluno pode ter algum controlo na simulação sendo:

* Controlo do ritmo e do progresso da simulação, pode avançar a instrução passo por passo, que faz com que a função continue mas de uma forma mais lenta ou então pode correr a função sem interrupções.
* Pode pausar a simulação, permitindo uma análise mais profunda dos dados (valor de uma variável) e/ou um debate com o professor ou com outros alunos.
* Permite retroceder passos da execução, para que o aluno possa repetir essas instruções para ajudar nas partes menos compreendidas pelos alunos nas execuções.

Durante a simulação, existe um componente que vai mudar de cor, para facilitar ao aluno o que se está a passar no programa. A Figura 3 mostra os aspetos do SICAS durante a simulação do algoritmo.

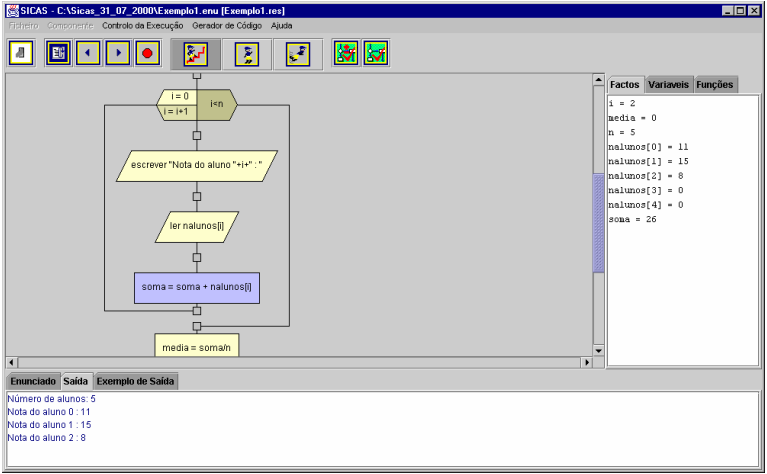


Figura 3 – Simulação do algoritmo

Se o professor tem dado um conjunto de dados de teste (entradas e correspondente resultado esperado do algoritmo) para o problema ser resolvido, o aluno também pode testar o seu próprio algoritmo com os mesmos dados. É útil se o professor especificar casos especiais dos dados de entrada, uma vez que muitos alunos tendem a ficar satisfeitos se as suas soluções funcionarem para o que é pedido sem se importar com outros casos.

**Conclusão**

O SICAS é uma aplicação educacional que tem intenção para ajudar o aluno a compreender e a tornar-se “profissional” na utilização de estruturas de programação básicas. O aluno pode desenhar/criar algoritmos que solucionam os problemas colocados pelo professor. O sistema usa uma interface gráfica interativa, onde as estruturas básicas de programação são representadas através de símbolos pré definidos.

Este ambiente pode ser usado pelos alunos em grupo ou em singular, nas aulas ou em casa. Permite aos estudantes com mais dificuldades (em acompanhar o ritmo nas aulas) de poder acompanhar as aulas praticando através da aplicação, permitindo assim uma melhoria da sua lógica e experiência.