ENSINO E APRENDIZAGEM DE ALGORITMIA COM A FERRAMENTA PORTUGOL IDE

António Manso[1], Célio Gonçalo Marques[2], Pedro Dias[3]

Abstract *— Computer programming is a subject which experiences high rates of educational failure. Computer programming is a complex activity involving high level of abstraction, which is usually taught resorting to programming languages and development environments directed towards software production within an industrial context. In an attempt to fight this failure rate, we have developed Portugol IDE (http:/www.dei.estt.ipt.pt/portugol/), a learning environment which uses an algorithm specification language named Portugol which is based on the students' mother tongue, a graphic language which reinforces the concept of sequential execution through flow diagrams and a set of debugging tools and animation intended to develop algorithmic reasoning. The present paper presents an introduction to Portugol IDE and some application results.*

Resumo *— A programação de computadores é uma área curricular que apresenta grandes níveis de insucesso. Esta é uma actividade complexa e com nível elevado de abstracção que, tradicionalmente, é ensinada recorrendo a linguagens de programação de uso geral e a ambientes de desenvolvimento projectados para a produção de software em contexto industrial. Como tentativa de combater este insucesso desenvolvemos o Portugol IDE (http:/www.dei.estt.ipt.pt/portugol/), um ambiente de aprendizagem de programação que possui uma linguagem de especificação de algoritmos baseada na língua materna dos alunos, o portugol, uma linguagem gráfica que reforça o conceito de execução sequencial, os fluxogramas, bem como um conjunto de ferramentas de depuração e animação de algoritmos que têm como objectivo principal o desenvolvimento do raciocínio algorítmico. Neste trabalho apresentamos a ferramenta Portugol IDE e alguns resultados da sua aplicação.*

Palavras-chave — Algoritmos, Ensino de programação, Fluxograma, Portugol.

INTRODUÇÃO

A programação de computadores é uma área curricular que é transversal a muitos cursos de engenharia, e obrigatória nos cursos de computação.

Esta área curricular apresenta, tradicionalmente, altos níveis de insucesso [1] e são muitos os factores que o influenciam.

Aprender uma nova linguagem cujo interlocutor é um computador, incapaz de resolver inconsistências, ou deduzir informação implícita, requer um novo tipo de comunicação e um novo tipo de expressão das mensagens: os algoritmos. Existem muitas e variadas linguagens de programação que permitem codificar os algoritmos, no entanto, a sua sintaxe e semântica é complexa e rígida. De modo a aliviar o esforço de representação dos algoritmos os docentes utilizam pseudo-linguagens na língua materna dos aprendentes (e.g. portugol) ou linguagens gráficas (e.g. fluxogramas) mais visuais e apelativas [2]. Este tipo de codificação tradicionalmente é feita em papel e a sua execução é feita por humanos, o que permite atenuar as regras sintácticas e semânticas. A verificação formal do algoritmo é feita através de testes de mesa o que a torna morosa, passível de erros, e pouco adequada aos hábitos tecnológicos das novas gerações de alunos.

Programar é uma actividade complexa, com altos níveis de abstracção, que requer maturidade intelectual para a sua execução. A programação aplica-se num conjunto alargado de problemas e contextos, e sendo uma área curricular de base é leccionada nos primeiros semestres dos cursos. Aprender a programar em contexto real é muito difícil, por isso, os alunos devem ser munidos de um conjunto de exemplos que ilustram e simulam pequenas partes das aplicações reais, desprovidas da complexidade do contexto original, e espera-se que estes adquiram um conjunto de técnicas que os irão ajudar a construir aplicações complexas no futuro.

As linguagens de programação utilizadas na produção de software são invariavelmente definidas no léxico inglês, o que dificulta a tradução do algoritmo para o programa com vista à sua execução e verificação automática.

1 António Manuel Rodrigues Manso, Departamento de Engenharia Informática, Escola Superior de Tecnologia de Tomar - Instituto Politécnico de Tomar, Quinta do Contador - Estrada da Serra, 2300-313 Tomar, Portugal, manso@ipt.pt, http://orion.ipt.pt/~manso/
2 Célio Gonçalo Cardoso Marques, Área Interdepartamental de Tecnologias de Informação e Comunicação, Escola Superior de Gestão de Tomar - Instituto Politécnico de Tomar, Quinta do Contador - Estrada da Serra, 2300-313 Tomar, Portugal, celiomarques@ipt.pt, http://www.celiomarques.com
3 Pedro Miguel Aparício Dias, Departamento de Engenharia Informática, Escola Superior de Tecnologia de Tomar - Instituto Politécnico de Tomar, Quinta do Contador - Estrada da Serra, 2300-313 Tomar, Portugal, pedrodias@ipt.pt, http://www.pedromdias.com

Os algoritmos são escritos num ambiente de desenvolvimento que foi projectado para o desenvolvimento de aplicações em contexto industrial. Os ambientes de desenvolvimento têm suporte para uma ou mais linguagens de programação, com um largo conjunto de ferramentas de auxílio à construção de programas de utilidade extrema na construção de software por programadores profissionais, mas demasiado complexas para a aprendizagem da programação.

Os factores identificados contribuem para altas taxas de reprovação e abandono dos alunos na área de computação [2]. O combate a este problema tem merecido grande atenção da comunidade académica e científica, através de novos métodos de ensino e ferramentas de auxílio á programação [3]-[4]-[5]-[8]-[9]-[10]-[11]. No âmbito deste esforço foi desenvolvido o Portugol IDE que utiliza o portugol e os fluxogramas de uma forma digital. A motivação dos alunos em utilizar ferramentas digitais é nitidamente superior ao uso de métodos tradicionais baseados em papel e aumenta o seu estudo autónomo [3]-[10].

PORTUGOL IDE

"*Pensar primeiro, programar depois*" é a máxima para a aprendizagem de programação de computadores. Esta abordagem requer que se desenvolvam ferramentas especificamente voltadas para o seu ensino e que permitam de uma forma simples a codificação do raciocínio algorítmico . Estas devem ser eficazes para a transmissão de conhecimentos por parte dos docentes e apelativas para os alunos explorarem os conceitos teóricos.

O Portugol IDE é um ambiente de aprendizagem de algoritmos que permite a codificação, execução e a depuração de algoritmos de uma forma agradável e intuitiva, focando a atenção do aluno no desenvolvimento do raciocínio algorítmico e não na linguagem de programação ou no ambiente de codificação. Os utilizadores deste ambiente são os docentes e alunos dos cursos introdutórios de programação, onde se espera que facilite o processo de ensino e aprendizagem. A aprendizagem da algoritmia necessita de muita prática para interiorizar os conceitos teóricos e enraizar a lógica computacional, e a ferramenta possui um conjunto de ferramentas que aumentam a autonomia do aluno fora dos períodos lectivos e na ausência dos docentes.

O Portugol IDE foi desenhado para ensinar os seguintes módulos de algoritmia: manipulação de informação, estruturas de decisão condicional e estruturas de iteração e repetição. Estes conceitos são implementados dentro do ambiente através de duas linguagens equivalentes entre si:

- Portugol - linguagem estruturada com sintaxe em português, com um léxico composto por poucas palavras reservadas, e com um conjunto simplificado de operadores. A linguagem não é sensível à capitulação e tem uma sintaxe flexível. Apesar da sua simplicidade, a linguagem é suficientemente poderosa para expressar algoritmos com raciocínios complexos;
- Fluxograma - linguagem gráfica composta por figuras geométricas que representam instruções computacionais e fluxos de dados que evidenciam a execução sequencial de instruções. A execução do fluxograma é feita com o auxílio da linguagem portugol.

O fluxograma é uma linguagem mais simples que o portugol e menos atreita a erros na codificação dos algoritmos, uma vez que este é desenhado através de formas parametrizáveis e a sua representação gráfica é mais apelativa e intuitiva. É, por isso, a indicada para os aprendentes darem os primeiros passos na unidade curricular de algoritmia.

TABELA I
INICIO E FIM DO ALGORITMO

Descrição	Portugol	Fluxograma
Início do algoritmo	`Inicio`	
Fim do algoritmo	`Fim`	

Manipulação de Informação

A base dos algoritmos são os dados que manipulam de forma a produzir informação útil ao utilizador. Os algoritmos são codificados entre as instruções de *inicio* e *fim* no portugol e por um símbolo parametrizavel no fluxograma (Tabela I). No fluxograma, ao primeiro símbolo a ser introduzido é-lhe automaticamente atribuído a valor "inicio" e o segundo o valor "fim". Se houver mais de dois símbolos deste tipo ou com os parâmetros iguais é assinalado um erro.

As linguagens tradicionais possuem um conjunto alargado de tipos dados com diversos tamanhos que são úteis na produção de software mas que introduzem uma dificuldade acrescida na aprendizagem. O portugol possui um conjunto de tipos de dados que representam apenas as categorias de dados independentemente do seu tamanho ou dos limites em que estão definidos (Tabela II.

TABELA II
TIPOS DE DADOS DO PORTUGOL

Descrição	Tipo	Valor por defeito
Número inteiro	`Inteiro`	0
Número real	`Real`	0.0
Valor lógico	`Lógico`	Falso
Caracter da tabela ASCII	`Caracter`	" " (espaço)
Conjunto de caracteres	`Texto`	"" (texto vazio)

A partir deste tipo de dados é possível definir variáveis cujo conteúdo pode ser manipulado e constantes cujo valor é definido na sua inicialização e permanece inalterado enquanto o símbolo estiver definido. A linguagem possui duas constantes pré-definidas (verdadeiro e falso) que servem de conjunto de suporte aos dados do tipo lógico.

O fluxograma não possui nenhum símbolo especial para a definição de dados. A definição de dados no fluxograma é

deduzida através dos símbolos de leitura e de cálculo de expressões (Tabela IV).

Para a definição de expressões computacionais o Portugol IDE possui os operadores que estão definidos na Tabela III, e que representam os operadores mais usuais na linguagem matemática. Para além dos operadores a linguagem possui um pequeno conjunto de funções matemáticas, cuja sintaxe podm ser consultadas na ajuda da linguagem.

TABELA III

OPERADORES

Tipo	Símbolos
Atribuição	`<-`
Aritméticos	`+ , - , *, / , % , ^`
Relacionais	`= , =/= , > , >= , < , <=`
Lógicos	`ou , e , não , xou`

A entrada de dados, o cálculo de expressões e a saída de informação são executados através das instruções e símbolos que estão expressos na Tabela IV.

Selecção de Instruções

O fluxograma possui um símbolo (losango) que permite fazer o desvio de fluxo de acordo com o valor lógico de uma condição. As instruções de decisão simples e em alternativa que estão presentes na linguagem portugol são feitas no fluxograma com o mesmo símbolo da forma mostrada na Tabela V.

TABELA IV

ENTRADA, SAÍDA E PROCESSAMENTO

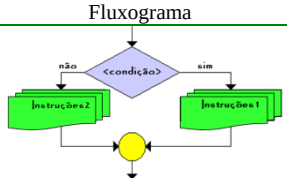
Portugol	Fluxograma
`Ler <variável>`	
`Escrever <expressão>, <...>`	
`Variável <- <expressão>`	

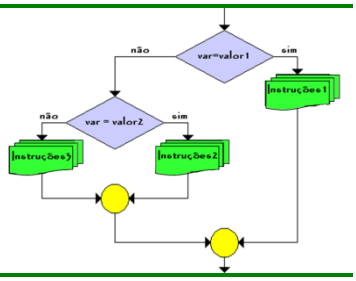
Esta representação permite ao aluno visualizar a execução da estrutura *escolhe-caso* como sendo um conjunto de decisões. Não foram introduzidas na linguagem as instruções de salto incondicional *break* e *continue* que, usualmente, estão presentes nas linguagens de alto nível, de forma a reforçar a execução sequencial e lógica de instruções.

TABELA V

ESTRUTURAS DE DECISÃO EM ALTERNATIVA

Portugol	Fluxograma

`Se <condição> então`
`  Instruções1`
`Senão`
`  Instruções2`
`Fim se`
	

`Escolhe <var>`
`  caso valor1:`
`  Instruções1`
`  caso valor2:`
`  Instruções2`
`  Defeito:`
`  Instruções3`
`Fim escolhe`
	

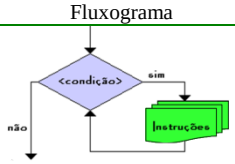
Repetição de Instruções

A linguagem portugol possui dois tipos de estruturas de repetição: a repetição de um conjunto de instruções com controlo inicial (Tabela VI) e a repetição com controlo final (Tabela VIII). O fluxograma representa estas estruturas com o auxílio do símbolo de decisão e o desvio de fluxo de para formar um ciclo de repetição.

TABELA VI

ESTRUTURA DE REPETIÇÃO ENQUANTO-FAZ

Portugol	Fluxograma

`Enquanto <condição> faz`
`  <Instruções>`
`Fimenquanto`
	

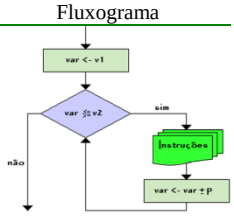
Ao contrário do que acontece com outros programas didácticos [10]-[12] optámos pela representação dos ciclos sem o auxílio de formas especializadas. Esta abordagem reforça o raciocínio algorítmico uma vez que são condições lógicas que controlam a repetição e o ciclo é visível pelas linhas de fluxo.

Uma das técnicas básicas que os alunos de programação devem adquirir é o conceito de iteração, onde um conjunto de instruções é executado um número de vezes controlado por um iterador. Na linguagem portugol esta iteração é executada pelo ciclo *para-próximo*, no fluxograma é executado pela estrutura definida na Tabela VII.

TABELA VII

ESTRUTURA DE REPETIÇÃO PARA-PRÓXIMO

Portugol	Fluxograma

`Para <var> de v1 ate v2 passo p`
`  Instruções`
`Próximo`
	

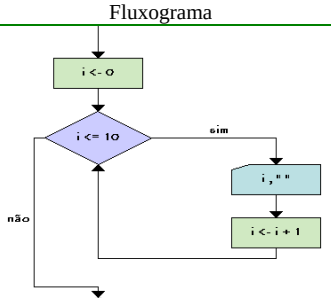
A estrutura mostra a equivalência entre o ciclo *enquanto-faz* e o ciclo *para-próximo* através do fluxograma, um exercício que usualmente é executado pelos aprendentes. As intruções da Tabela VII foram dotadas de tecnologia inteligente: o interpretador verifica os limites de iteração e

automaticamente adapta o operador de comparação (>= ou <=) da decisão que controla o ciclo e o operador aritmético (+ ou -) da instrução que actualiza a variável de iteração. O parâmetro *passo* da instrução *para-próximo* pode ser omitido e por defeito é-lhe atribuído o valor unitário. Esta tecnologia permite que o aprendente expresse os ciclos de iteração numa linguagem próxima da natural através do portugol. A Tabela VIII mostra a utilização desta tecnologia num algoritmo que escreve os números inteiros de 0 e 10.

TABELA VIII

ALGORITMO PARA ESCRITA DOS NÚMEROS INTEIROS NO INTERVALO [0,10]

Portugol	Fluxograma

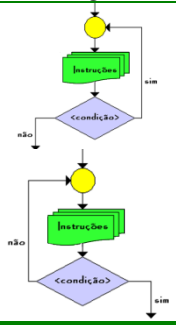
`inteiro i`
`para i de 0 ate 10`
`  escrever i , " "`
`proximo`
	

A Tabela IX possui a definição das estruturas de repetição com controlo final. Estas estruturas são úteis quando o algoritmo necessita que um bloco de instruções seja executado pelo menos uma vez. As duas instruções diferem apenas no desvio de fluxo através da condição que controla o ciclo.

TABELA IX

ESTRUTURA DE REPETIÇÃO FAZ ENQUANTO

Descrição	Fluxograma

`Faz`
`  Instruções`
`Enquanto <condição>`
 	
`Repete`
`  Instruções`
`Até <condição>`
	

AMBIENTE DE APRENDIZAGEM

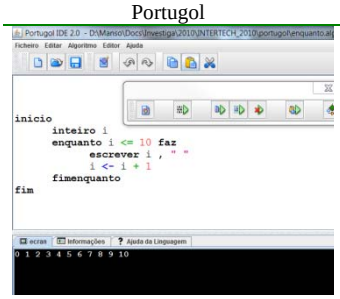
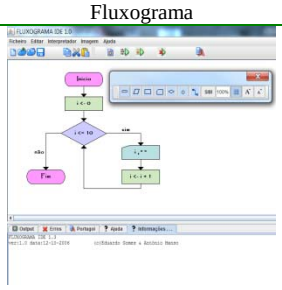
O ambiente Portugol IDE permite que os aprendentes expressem os algoritmos na linguagem que julgarem mais conveniente (escrita ou gráfica), e a tecnologia implementada no sistema faz a tradução para a representação complementar. Ambas as representações permitem fazer a verificação sintáctica e semântica do algoritmo bem como a sua execução e depuração.

A depuração do algoritmo permite que o aprendente visualize as instruções computacionais que estão a ser executadas e a forma como essas instruções afectam as varáveis definidas na memória. Na depuração do programa em portugol a instrução a ser executada é realçada através da alteração da cor de fundo e no fluxograma através da alteração da cor da forma a ser executada e do realce do fluxo de execução.

O portugol permite definir variáveis em qualquer parte do programa. Esta característica permite ensinar aos alunos o escopo das variáveis, bem como a criação e destruição de variáveis dentro de um bloco de instruções. As varáveis na memória adquirem uma cor de acordo com o nível do bloco em que estão definidas.

TABELA IX

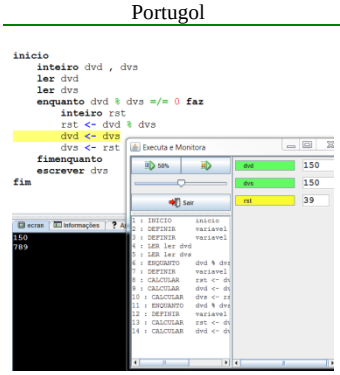
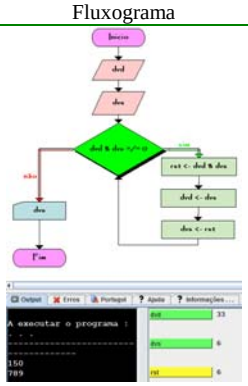
PORTUGOL IDE - DUAS VISÕES DO MESMO ALGORITMO

Portugol	Fluxograma
	

Na Tabela X a variável *rst* está definida dentro do ciclo o que faz com que seja criada e destruída em cada iteração. Esta variável adquire a cor amarela (nível 2) enquanto as outras duas adquirem a cor verde (nível 1).

TABELA X

DEPURAÇÃO DO ALGORITMO DE EUCLIDES

Portugol	Fluxograma
	

RESULTADOS PEDAGÓGICOS

O Portugol IDE tem sido utilizado nos cursos de engenharia, e nos cursos de especialização tecnológica da Escola Superior de Tecnologia de Tomar do Instituto Politécnico de Tomar desde 2005. A evolução do sistema reflecte as necessidades dos docentes para ensinar os conceitos teóricos e dos alunos que regularmente são

convidados a fazer sugestões sobre a sua evolução. Esta simbiose entre as duas partes do processo de ensino-aprendizagem faz com o Portugol IDE seja hoje uma ferramenta que é utilizada e acarinhada pela comunidade académica.

Num estudo efectuado com alunos do 1º ano da licenciatura em Engenharia Informática [5] constatou-se que a ferramenta Portugol IDE é fácil e agradável de utilizar e fácil de lembrar. Na opinião dos alunos o módulo de escrita e execução em fluxograma, a visualização do estado das variáveis a ajuda na correcção de erros são úteis. Os alunos referem ser fácil codificar algoritmos e corrigir erros quando é usada esta ferramenta.

Esta ferramenta reforça o desenvolvimento de competências pelos alunos tornando-os mais responsáveis pelo processo de ensino e aprendizagem. Procura também acompanhar a evolução tecnológica e as necessidades e hábitos das actuais gerações de alunos. Consideramos também que esta ferramenta intensifica a actividade cognitiva dos alunos e potencia a sua autonomia no estudo.

CONCLUSÃO

Programar é uma actividade muito estimulante e intelectualmente gratificante desde que se domine a lógica de programação e se possua os conhecimentos de uma linguagem de programação. O Portugol IDE pretende estimular o desenvolvimento da lógica algorítmica e fazer uma transição suave para as linguagens de programação de alto nível.

A definição das linguagens de representação de algoritmos com poucas instruções e formas facilita a sua aprendizagem. Esta simplicidade não impede, no entanto, que estas linguagens sejam suficientemente poderosas para a construção de algoritmos complexos. As ferramentas disponibilizadas dentro do ambiente do Portugol IDE permitem que os alunos codifiquem, executem e façam a verificação formal de algoritmos de uma forma intuitiva e pedagogicamente estimulante.

Os alunos da Escola Superior de Tecnologia de Tomar do Instituto Politécnico de Tomar estão bastante satisfeitos com a ferramenta e consideram-na de grande utilidade para a aprendizagem de algoritmos.

Motivados pelo êxito pedagógico do Portugol IDE, está em desenvolvimento a versão 3.0 da ferramenta que suportará a definição de dados complexos (matrizes e registos) e terá suporte para funções iterativas e recursivas. Está também em desenvolvimentos a integração da ferramenta com um repositório de algoritmos que permitirá a construção e correcção automática de exercícios, bem como a construção de um agente inteligente para auxílio á correcção lógica do algoritmo. Estas novas características visam a promoção do estudo autónomo da programação e o aumento das técnicas de programação suportadas.

BIBLIOGRAFIA

[1] Butler, M., & Morgan, M., Learning challenges faced by novice programming students studying high level and low feedback concepts. In *ASCILATE* 2007 Singapore, 2007, pp. 99-107.

[2] Esmin, A. A. A., Portugol/Plus: Uma ferramenta de apoio ao ensino de lógica de programação baseado no portugol, *IV Congresso RIBIE*, Brazilia, 1998.

[3] Gondim, H. W.& Ambrósio, A. P., Esboço de Fluxogramas no Ensino de Algoritmos, *Anais do XXVIII Congresso da SBC*, 2008.

[4] Jesus, E. A, Ferramenta para criação e teste de algoritmos utilizando Fluxogramas ou Portugol, *IV Congresso Brasileiro de Computação – CBComp 2004*, 2004, pp. 213-218.

[5] Manso, A., Oliveira L. & Marques, C. G. C., Ambiente de Aprendizagem de Algoritmos – Portugol IDE. In P. Dias & A. Osório (Orgs.), *Actas da VI Conferência Internacional de TIC na Educação (Challenges 2009)*. Braga: Centro de Competência da Universidade do Minho, 2009, pp. 969-983.

[6] Manso, A., Oliveira L., & Marques, C. G. C., Ensino da Programação através da Linguagem Algorítmica e Fluxográfica. In A. Rocha, F. Restivo, L. P. Reis & S. Torrão (Orgs.), Sistemas de Tecnologias de Informação. *Actas da 4.ª Conferência Ibérica de Sistemas e Tecnologias de Informação (CISTI). WCO&NTiHE - Workshop on Challenges, Opportunities and New Trends in Higher Education*. Porto: Associação Ibérica de Sistemas e Tecnologias de Informação, Faculdade de Engenharia da Universidade do Porto e Universidade Fernando Pessoa, 2009.

[7] Manso, A., Oliveira L.,& Marques, C. G. C. , Portugol IDE – A Tool for Teaching Programming. In D. Carvalho, N. V. Hattum-Janssen & R. M. Lima (Orgs.), *1st Ibero-American Symposium on Project Approaches for Engineering Education (PAEE' 2009).* Guimarães: Research Centre in Education (CiEd) of University of Minho and Department of Production and Systems of School of Engineering of University of Minho, 2009, pp. 95-101.

[8] Paes, A. & Almeida, E., Gerando Pseudo-código a partir de Funções computáveis utilizando uma ferramenta de simulação, *Simposio Brasileiro de Informática na Educação*, Fortaleza, 2008.

[9] Santiago, R., Dazzi, R.,Rudimar L.S., Ferramenta de apoio ao ensino de algoritmos", *Seminário de Computação*, Blumenau, 2004.

[10] Vargas, K. S., & Martins, J., Ferramenta para Apoio ao Ensino de Introdução à Programação, *XIV Seminário de Computação*, Universidade Regional de Blumenau, Santa Catarina, 2005.

[11] Valadares, C. M. S., Ensino a distência de Linguagens de programação de Computadores, *15º Congresso internacional ABED de educação*, Fortaleza, CE, 2009.

[12] Marcelino, M., Mihaylov, T., & Mendes, A., H-SICAS, a Handheld Algorithm Animation And Simulation Tool To Support Initial Programming Learning - *38th ASEE/IEEE Frontiers in Education Conference*. Saratoga Springs, NY, 2008.