



UNIVERSIDADE ESTADUAL DO CEARÁ - UECE
Centro de Ciências e Tecnologia - CCT

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DO CEARÁ
CEFET - CE

Diretoria de Pesquisa e Pós-graduação – DIPPG



MESTRADO INTEGRADO PROFISSIONAL EM COMPUTAÇÃO APLICADA
UECE / CEFET – CE

FRANCISCO EDMAR VASCONCELOS PEREIRA

ANÁLISE DE UM SOFTWARE EDUCATIVO PARA
MEDIAÇÃO DO ENSINO DE POO

FORTALEZA - CEARÁ

2007

FRANCISCO EDMAR PEREIRA VASCONCELOS

ANÁLISE DE UM SOFTWARE EDUCATIVO PARA
MEDIÇÃO DO ENSINO DE POO

Dissertação apresentada ao Curso de Mestrado Profissional em Computação Aplicada da Universidade Estadual do Ceará e do Centro Federal de Educação Tecnológica do Ceará como requisito parcial necessário para obtenção do título de Mestre em Computação Aplicada.

Orientadora: Prof(a). D.Sc. Verônica Lima Pimentel de Sousa

FORTALEZA - CE

2007

V331a

PEREIRA, Francisco Edmar Vasconcelos

Análise de um software educativo para mediação do ensino de POO./ Francisco Edmar Vasconcelos Pereira. Fortaleza: UECE/CEFETCE, 2007.

89p.

Dissertação (Mestrado Profissional em Computação Aplicada)

Orientadora: D. Sc. Verônica Lima Pimentel de Sousa

1. INFORMÁTICA EDUCATIVA 2. PROGRAMAÇÃO ORIENTADA A OBJETOS – SOFTWARE EDUCATIVO – AVALIAÇÃO 3. AMBIENTES INFORMATIZADOS DE APRENDIZAGEM I. Título

CDD – 005.1310285

Título do Trabalho: Análise de um Software Educativo para Mediação do Ensino de POO

Autor: Francisco Edmar Pereira Vasconcelos

Defesa em: ___/___/___

Conceito Obtido: _____

BANCA EXAMINADORA:

Prof. D. Sc. Verônica Lima Pimentel de Sousa
Orientador(a) - Centro Federal de Educação Tecnológica do Ceará

Prof. D. Sc. Mauro Cavalcante Pequeno
Membro externo – Universidade Federal do Ceará

Prof. PhD. Elian de Castro Machado
Membro interno – Universidade Estadual do Ceará

Aos meus pais,
Ismar Vasconcelos Pereira e
Francisco Edmar Pereira (*in memoriam*).

“O que sabemos é uma gota, o que ignoramos
é um oceano”. (Isaac Newton)

AGRADECIMENTOS

Agradeço à minha orientadora, profa. Dra. Verônica Pimentel, pela capacidade visionária de criar condições para que o objeto desta dissertação existisse.

À profa. Dra. Cassandra Ribeiro, pelo diálogo, pelas sugestões teóricas, pelas correções de rumo nos momentos iniciais deste trabalho.

Aos membros da banca examinadora, professores Mauro Pequeno e Elian Machado, pela leitura deste trabalho e pelas críticas e sugestões no momento da defesa visando enriquecer este texto.

Aos professores da área de programação da Gerência de Telemática do CEFET-CE, especialmente aos que participaram da avaliação do software educativo do objeto desta dissertação, que me deram subsídios para vislumbrar novos caminhos no ensino da programação.

Ao CEFET-CE, pelo incentivo e apoio financeiro que tornou possível este trabalho.

Aos professores do MPCOMP, pelas condições oferecidas e ensinamentos.

Ao amigo Régis Conde de Moura pelas revisões feitas neste trabalho.

À Adriana Maciel, pela parceria constante e colaboração imprescindível.

Aos meus filhos Edmar Neto e Érica, minha gratidão por acreditarem sempre que “tudo vai dar certo”.

RESUMO

O ensino de programação tem recebido ênfase nos últimos anos devido a utilização do novo paradigma de Orientação a Objetos (OO), cujo o impacto reflete tanto o mercado como a formação de programadores. Vários problemas com o ensino e a utilização do conceito de OO têm sido identificados e discutidos na literatura, e soluções para estes problemas vêm sendo largamente estudadas. Softwares educativos, de um modo geral, vêm ajudando aos professores a tornar mais eficiente o processo de ensino e, conseqüentemente, o aprendizado. Neste trabalho, discutiu-se o ensino de POO sob a ótica da mediação pedagógica. Para tanto, apresentamos inicialmente o conceito de mediação em seu sentido amplo e, posteriormente, caracterizamos, sobre bases teóricas, o significado de “mediação pedagógica”. Após a devida fundamentação, analisamos o software educativo Karel J. Robot. Este estudo objetiva a análise e caracterização do potencial e recursos do Karel J. Robot. Uma metodologia específica foi utilizada e não experiências observadas em sala de aula, pretendendo-se observar o uso por parte de professores, do software específico de OO na mediação pedagógica. Participaram da avaliação do software 8 (oito) professores de POO do CEFET-CE. A análise foi bem sucedida na avaliação da mediação pedagógica do software Karel J. Robot. Os dados obtidos mostram não só a adequação como a eficiência pedagógica do software como recurso de mediação para professores de POO.

PALAVRAS-CHAVE: Informática Educativa, Mediação Pedagógica, Programação Orientada a Objetos, Software Educativo.

ABSTRACT

The teaching of Object-Oriented Programming (OOP) has received great emphasis in the last years due to the use of the new paradigm that it impacts both in the market and the programmers' formation, the Object-Oriented (OO) paradigm. Many problems with the teaching and use of the OO concept have been identified and discussed in literature, and solutions to these problems have been widely studied. Educational software, in general, has been helping teachers to make the teaching process easier and, consequently, the learning. This work has discussed OOP teaching under the pedagogical mediation optics. In order to do so, we have initially presented the concept of mediation in its broad sense and, afterwards, we have characterized the meaning of "pedagogical mediation" under theoretical bases. Then, we have analyzed the Karel J. Robot pedagogical software. This study aims the analysis and characterization of the Karel J. Robot potential and resources regarding the pedagogical mediation. A specific methodology and not a classroom experiences observation of the specific OO software have been used in the pedagogical mediation. Eight professors from CEFET-CE who works with OOP were involved in the evaluation of that software. The analysis was well succeeded in the evaluation of the Karel J. Robot software pedagogical mediation. The collected data show not only the adequacy, but also the pedagogical efficiency of the software as a mediation resource for OOP professors.

KEY WORDS: Informatics for Education, Pedagogical Mediation, Object-Oriented Program, Educational Software.

LISTA DE ILUSTRAÇÕES

FIGURA 1 – INTERFACE GRÁFICA DO KAREL J. ROBOT.....	20
FIGURA 2 – DIAGRAMA DE CLASSE URROBOT.....	21
FIGURA 3 – DIAGRAMA DA HIERARQUIA DE CLASSE DO KAREL J. ROBOT.....	21
FIGURA 4 – O MUNDO DO KAREL J. ROBOT.....	22
FIGURA 5 – SITUAÇÃO INICIAL.....	23
FIGURA 6 – SITUAÇÃO FINAL.....	24
FIGURA 7 – CÁLCULO SOMA SITUAÇÃO INICIAL E FINAL.....	25
FIGURA 8 – SUBIR DEGRAUS SITUAÇÃO INICIAL E FINAL.....	25
FIGURA 9 – CORRIDA COM OBSTÁCULOS SITUAÇÃO INICIAL E FINAL.....	26
FIGURA 10 – ESCAPAR DE LABIRINTOS: SITUAÇÕES INICIAL E FINAL.....	26

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

AT&T – American Telephone and Telegraph

CAI – Computer Assisted Instruction

CRC – Class Responsibility and Collaboration

DARPA – Defense Advanced Research Projects Agency

FTP – File Transfer Protocol

HTML – Hypertext Markup Language

IRC – Internet Relay Chat

LOO – Linguagem Orientada a Objetos

MAEP – Método Ergopedagógico Interativo de Avaliação para Produtos Educacionais
Informatizados

NSF – National Science Foundation

OO – Orientação a Objetos

OOD – Object Oriented Design

POO – Programação Orientada a Objetos

RDD – Responsibility-Driven Design

UML – Unified Modelling Language

WWW – World Wide Web

SUMÁRIO

LISTA DE ABREVIATURAS E SIGLAS.....	XII
<u>1 INTRODUÇÃO.....</u>	<u>XIII</u>
1.1 PROBLEMÁTICA DO ENSINO DE POO E ESTRATÉGIAS APLICÁVEIS.....	XV
1.2 OBJETIVO GERAL.....	XVI
1.3 OBJETIVOS ESPECÍFICOS.....	XVI
1.4 ESTRUTURA DESTE TRABALHO.....	XVII
<u>2 O SOFTWARE KAREL J. ROBOT.....</u>	<u>.19</u>
1.5 A INTERFACE DO SOFTWARE.....	19
1.6 A HIERARQUIA DE CLASSES DO KAREL J. ROBOT.....	20
1.7 O MUNDO DO ROBÔ.....	22
1.8 SITUAÇÕES DE PROGRAMAÇÃO DO ROBÔ.....	23
REFERÊNCIAS BIBLIOGRÁFICAS.....	28

1 INTRODUÇÃO

O conceito de Programação Orientada a Objetos (POO) não é novo. No final da década de 60, a linguagem Simula67, desenvolvida na Noruega, introduzia conceitos hoje encontrados nas Linguagens Orientadas a Objetos (LOO). Em meados de 1970, o centro de pesquisa da Xerox (PARC) desenvolveu a linguagem Smalltalk, a primeira totalmente orientada a objetos. No início da década de 80, a AT&T lançaria C++, uma evolução da linguagem C voltada à orientação a objetos. Atualmente, a grande maioria das linguagens como Java, Object Pascal, C++, etc., incorpora características do paradigma Orientado a Objetos (OO).

A utilização deste paradigma vem crescendo consideravelmente tanto em nível comercial quanto em nível acadêmica sendo a preocupação com o ensino uma realidade mundial, recebendo assim, por parte de educadores e pesquisadores, cada vez mais atenção.

Para propiciar melhores formas de comunicação da informação e de sua compreensão, o grande desafio lançado à educação, hoje em dia, é o de usar as novas tecnologias existentes como ferramenta de ensino.

No ensino da programação, obrigatoriamente iremos usar o computador como ferramenta, pois ele é o próprio instrumento de trabalho. Assim se faz necessário justificar sua adequação ao propósito deste estudo: atuar eficientemente no ensino de POO.

Os programas educacionais podem fazer do computador uma ferramenta verdadeiramente eficaz à aprendizagem, sendo que o software educativo pode ser visto como um excelente instrumento na solução de problemas de aprendizagem. O software educativo:

... Trabalha com representações virtuais de forma coerente, mas extremamente flexível, possibilitando a descoberta e criação de novas relações. Dispõe suas informações de forma clara, objetiva e lógica, facilitando a autonomia do usuário, favorecendo a exploração espontânea. Exige também que o usuário tenha consciência do que quer, se organize e informe de modo ordenado o que fazer digitando corretamente. Dá um retorno extremamente rápido e objetivo do processo em construção, favorecendo a autocorreção, a inserção da “desordem” na ordem global. Trabalha com uma disposição espacial das informações, que pode ser controlada continuamente pelo usuário através de campo perceptivo visual, apoiando o raciocínio lógico. (OLIVEIRA, 1998: p. 126)

Portanto, sendo o computador um instrumento programado de forma lógica e simbólica favorece o desenvolvimento de atividades mentais levando o aluno à descoberta e invenção de relações entre as informações, ou seja, a organizar sistemas de forma coerente, clara e criativa.

Com o desenvolvimento das ferramentas computacionais e sua inserção nos meios educacionais, surgem novos instrumentos didáticos que devem ser utilizados pelos professores das disciplinas que contemplem o paradigma OO. Estes se vêem diante do desafio de experimentar novas formas de ensinar com o auxílio do computador.

MENDES (2000) já alertava dizendo que aprender a programar é um processo reconhecidamente difícil para a maioria dos estudantes. Mesmo levando-se em conta a aprendizagem dos conceitos básicos e sua aplicação na resolução de problemas concretos, existe um alto grau de dificuldade à compreensão destes. Ele afirma que a consciência destas dificuldades tem levado educadores de todo mundo a procurar estratégias e materiais que

possam ajudar a minorar as dificuldades sentidas por muitos estudantes e professores de programação.

1.1 Problemática do Ensino de POO e Estratégias Aplicáveis

Várias questões sobre a metodologia de ensino e aplicação do paradigma OO, foram levantadas com o objetivo de se obter um melhor proveito de seus benefícios, tais como representação do mundo real de forma simples, reuso, facilidade de manutenção etc. (RUMBAUGH, 1994).

DE CLUE (1996) aponta como uma das principais dificuldades para o aprendizado de OO a diferença de abordagem do problema. Enquanto o paradigma Estruturado tem como principal foco as ações (procedimentos e funções) o paradigma OO se preocupa com os objetos e seus relacionamentos. Por outro lado, além do conceito de objeto o paradigma OO, segundo a maioria dos especialistas, centra-se em quatro princípios: abstração, encapsulamento, herança e polimorfismo, cujos conceitos podem não ficar totalmente claros para o aluno se forem abordados de forma inadequada.

ALVAREZ (1995) coloca outras questões com relação ao ensino de OO, como: quando introduzir o paradigma OO, quais os conceitos incluir em um curso introdutório, qual metodologia usar para apresentar os conceitos do paradigma para que sejam assimilados de forma adequada, quando ensinar a primeira linguagem de programação, qual a linguagem de programação a ser usada etc., pois dependendo da abordagem adotada em relação a estas questões pode-se chegar a diferentes resultados em relação ao ensino do paradigma como um todo. Caso sejam feitas escolhas inadequadas, isto pode levar a problemas de ensino-aprendizagem e como consequência, à má utilização do paradigma.

BECKER (1998) enfatiza que os estudantes devem adquirir habilidades de decompor sistemas em classes de objetos, de definir as operações de cada classe e de identificar os possíveis relacionamentos existentes entre as mesmas. Para isso, se faz necessário a quebra de paradigma, ou seja, adquirir o pensamento orientado a objetos (*Object – Oriented thinking*).

Essas considerações nos levam a entender que, a eficácia do ensino do paradigma OO está em auxiliar o aprendiz a propor soluções em termos de objetos, classes, troca de mensagens, etc., requerendo para isto mais do que o uso de uma linguagem de POO.

Assim, considera-se necessário pesquisar, identificar e avaliar ferramentas de mediação pedagógica, tal que maximizem a eficácia do ensino de POO.

Pesquisadores e educadores da área de ensino de programação têm reunido esforços e desenvolvem ferramentas para auxiliar a assimilação desses conceitos pelos estudantes.

Um exemplo é o software educativo Karel J. Robot, desenvolvido em Mellon University – NY, que consiste em um método para ensino da linguagem Java podendo ser usado no primeiro dia de aula tendo como público alvo alunos que não têm nenhuma experiência com programação, ou com o paradigma OO.

Isto posto, este trabalho tem caráter exploratório com a finalidade de pesquisar e avaliar os recursos do software educativo Karel J. Robot, e limita-se às possibilidades de verificação da eficácia pedagógica para mediação do processo ensino-aprendizagem dos conceitos de POO.

Como já mencionado, no ensino da programação o computador é o próprio instrumento de trabalho. Deste modo, o recurso usado para a ação pedagógica fica restrito ao computador como recurso tecnológico e não como mediador do ensino, portanto, perdendo a potencialidade deste recurso para a ação pedagógica.

Este trabalho se concentra em apresentar o ensino da POO, através da exploração de recursos de novas tecnologias para mediação pedagógica.

1.2 Objetivo Geral

Verificar os recursos do software Karel J. Robot como instrumento de mediação para o processo ensino-aprendizagem de POO.

1.3 Objetivos Específicos

Os objetivos específicos deste trabalho podem ser resumidos da seguinte maneira:

- Contextualizar as principais teorias da aprendizagem e suas inter-relações com o ensino mediado por ambientes de aprendizagem computadorizados.

- Identificar, a partir da literatura e da prática pedagógica, as metodologias para o ensino de POO.
- Explorar os recursos do software Karel J. Robot para utilização do ensino de POO.
- Avaliar o referido software sob o aspecto da mediação pedagógica.
- Obter subsídios que justifiquem a utilização deste software, como recurso didático, para disciplinas de POO.

1.4 Estrutura deste Trabalho

O presente trabalho está estruturado da seguinte forma:

O Capítulo 2 apresenta a primeira parte da revisão de literatura que trata da influência da tecnologia na aprendizagem e mediadores das relações do ensino. Apresenta o histórico da introdução do computador no ensino, seu papel na relação ensino-aprendizagem e suas vantagens em relação aos outros meios.

O Capítulo 3 apresenta a segunda parte da revisão da literatura que discute a classificação dos softwares educacionais de acordo com suas características. Apresenta ainda as características de alguns softwares utilizados para o ensino de POO, específicos ou não. Analisa os serviços da Internet e por fim, as formas de avaliar softwares educacionais.

O Capítulo 4 apresenta, em detalhes, as características técnicas do software, específico para o ensino de POO, Karel J. Robot, bem como descreve a interface e explica as funções dos diversos comandos disponíveis.

O Capítulo 5 tem como principal enfoque explicar a metodologia utilizada para esta avaliação, cuja característica principal foi a adaptação do método de avaliação ergo - pedagógica – MAEP para produtos educacionais informatizados.

O Capítulo 6 descreve um estudo de caso da avaliação do software Karel J. Robot na abordagem do MAEP com o grupo de professores. Apresenta tabelas com os resultados da avaliação, bem como discute os mesmos.

O Capítulo 7 apresenta as conclusões desta pesquisa e recomenda estudos para trabalhos futuros.

2 O SOFTWARE KAREL J. ROBOT

Como foi mencionado no Capítulo 3, seção 3.2, as ferramentas de micro-mundos são utilizadas como apoio a processo de ensino em diversas áreas. O Karel J. Robot pertence a este grupo, por atuar no processo de ensino, especificamente em OO.

A primeira versão do software que descrevemos neste capítulo, denominada Karel the Robot, foi escrita em 1981 para auxiliar no ensino da linguagem Pascal que consistia em um método para ensino desta linguagem. O Karel++, 1997, foi concebido com a finalidade de apresentar a POO em C++ para iniciantes. Em 2001 foi apresentada a versão Karel J. Robot (BERGIN 2005), mantendo os mesmos padrões pedagógicos das versões anteriores, porém com uma sintaxe puramente Java.

1.5 A Interface do Software

A Figura 6 mostra a janela de interface gráfica do Karel J. Robot e a janela de controle na qual o aprendiz pode manipular ações do programa como: executar o programa e aumentar ou diminuir a velocidade de execução.

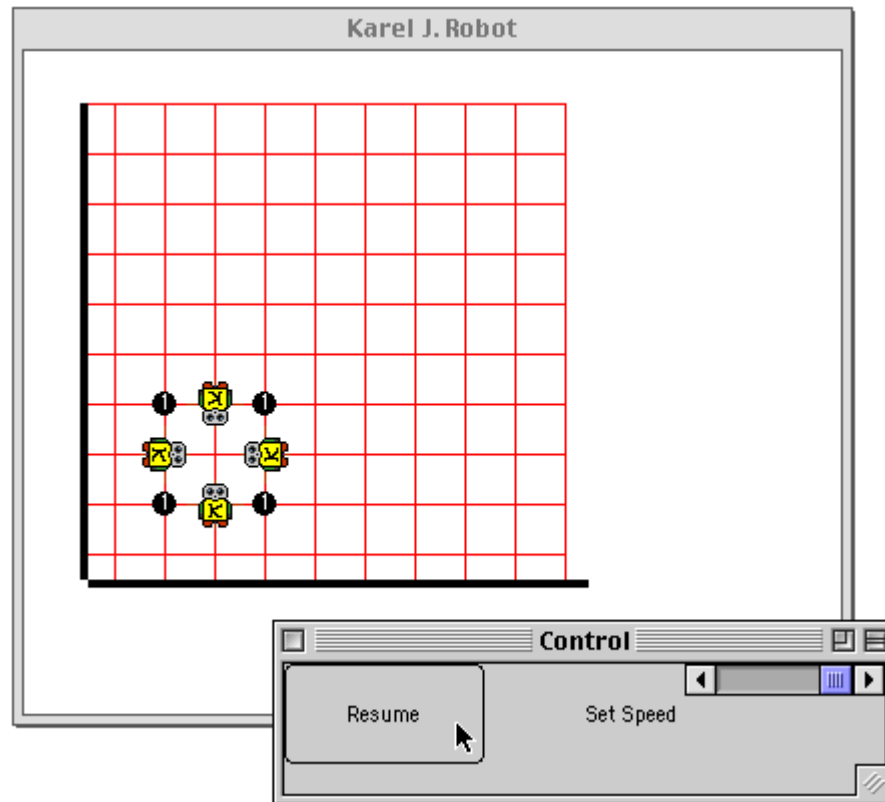


Figura 1 – Interface gráfica do Karel J. Robot

No Karel J. Robot o estudante tem contato com uma classe primitiva de robôs (Objeto) que é capaz de executar operações básicas como pegar e soltar objetos, andar, girar para a esquerda e se auto-desligar. Essa classe tem como atributos a localização e direção do robô, além do número de peças que este pode carregar.

1.6 A Hierarquia de Classes do Karel J. Robot

A Figura 7 apresenta o diagrama de classe UML – Unified Modeling Language, da classe UrRobot que contém os atributos e operações acima referidos. Nota-se também que o atributo direction (direção) é derivado da classe Directions.

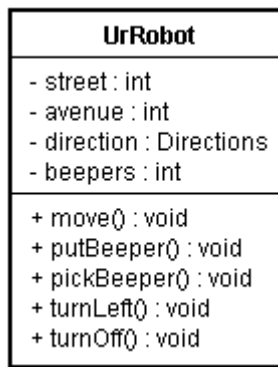


Figura 2 – Diagrama de classe UrRobot

Existe outra classe nativa, porém mais sofisticada que permite verificar se o robô está próximo de outros objetos como: outros robôs, peças ou seções de paredes, para qual direção está apontando, se a frente está livre, etc.

A Figura 8 apresenta o diagrama de classe Robot onde estão definidas estas funcionalidades, como também pode-se observar que a classes Robot estende a classe UrRobot.

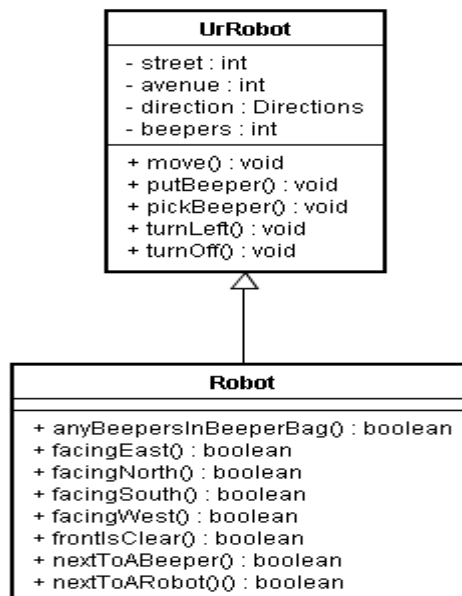


Figura 3 – Diagrama da hierarquia de classe do Karel J. Robot

1.7 O mundo do robô

O mundo do robô, figura 9, é representado graficamente como um plano cartesiano constituído por "ruas" e "avenidas" por onde pode se deslocar. É limitado ao Sul e ao Oeste por barreiras intransponíveis. Um robô sempre ocupa uma esquina, cruzamento de uma rua com uma avenida, e a origem é a esquina da rua 1 com a avenida 1. Neste mundo o robô tem contato com outros objetos como: seções de paredes, sinalizadores ou outros robôs.

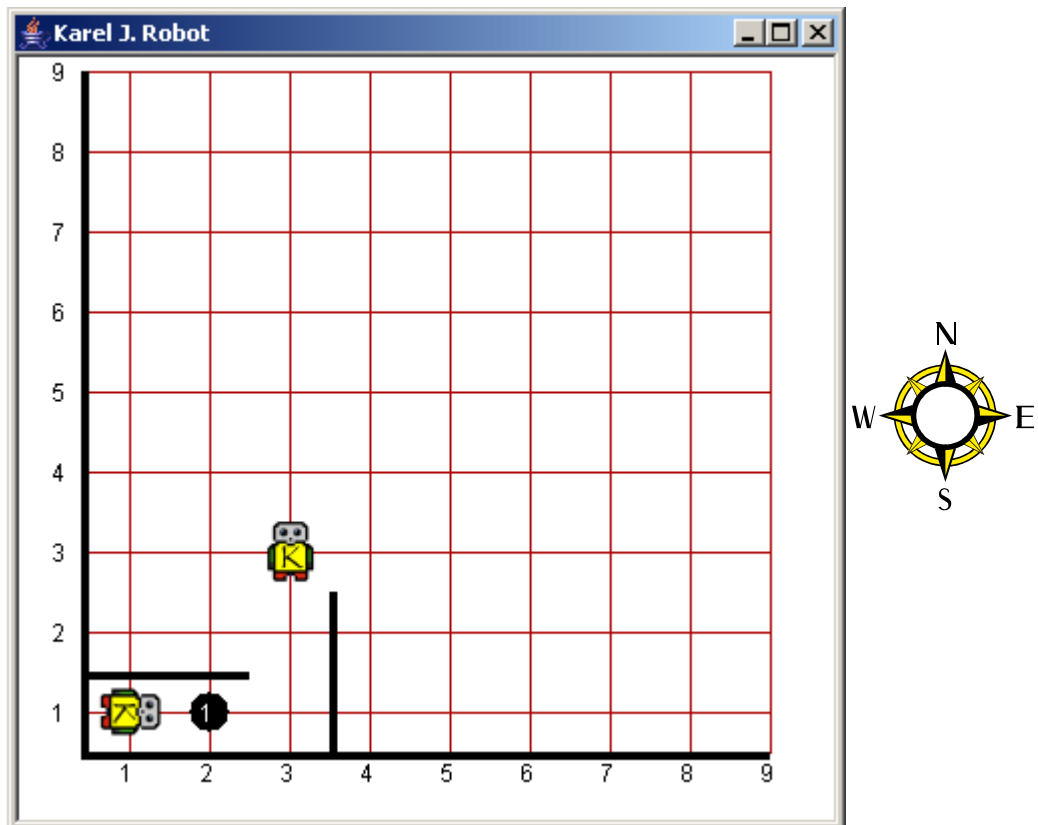


Figura 4 – O mundo do Karel J. Robot

Todas as mensagens enviadas aos robôs solicitando que estes executem serviços têm um retorno visual para o aluno. Por exemplo: se o robô recebe uma mensagem para pegar um objeto, esse objeto desaparece da tela. Se uma determinada instância de robô recebe uma mensagem para se deslocar o robô se movimenta na tela.

1.8 Situações de programação do robô

Conforme pode-se observar, na Figura 10 e Figura 11, o robô foi instruído a pegar um sinalizador que está na esquina da rua 1 com a avenida 2, situação inicial, e colocá-lo na esquina da rua 3 com a avenida 3, deslocar-se uma posição na direção norte e se desligar, situação final.

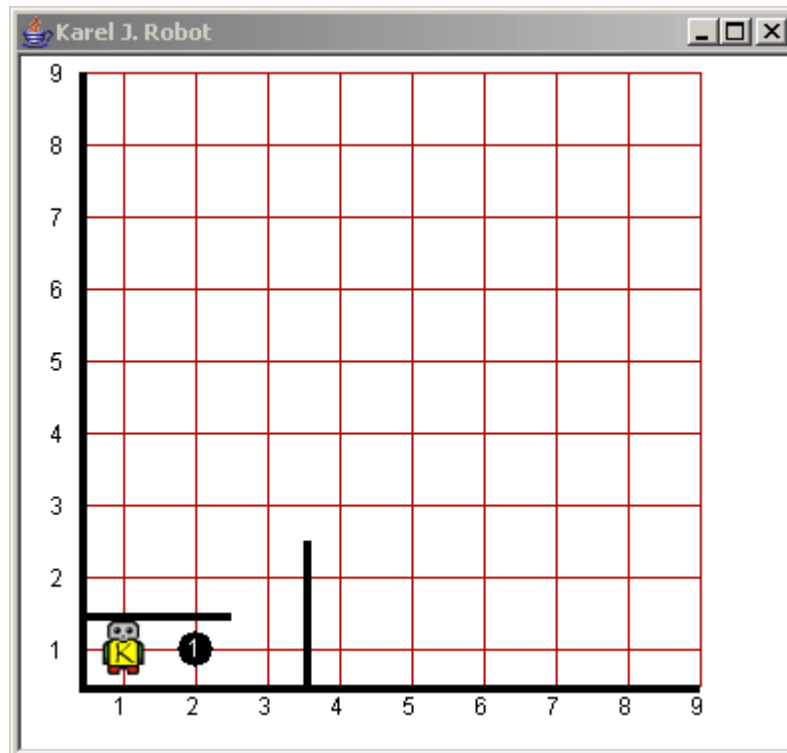


Figura 5 – Situação Inicial

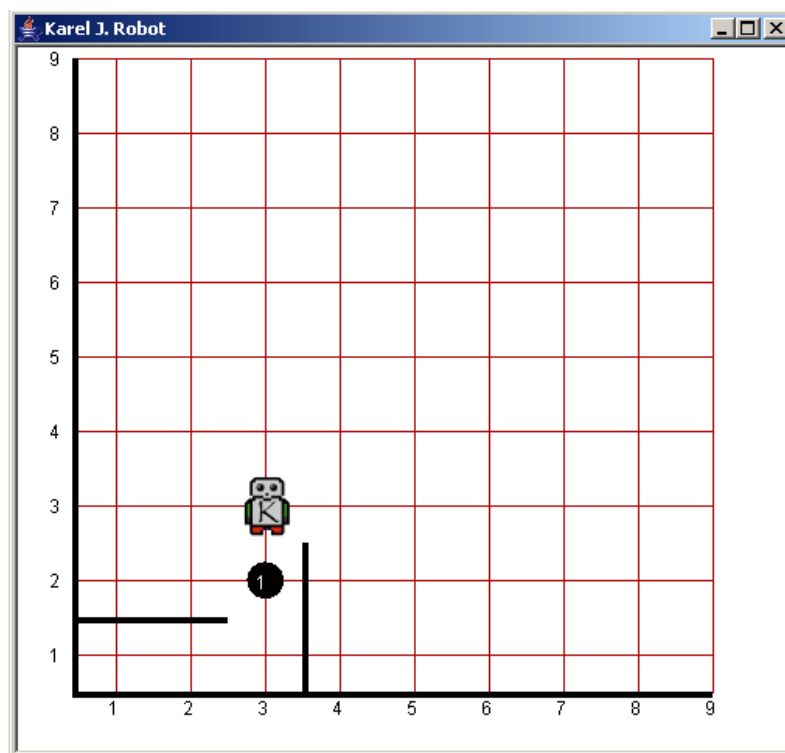


Figura 6 – Situação final

O Karel J. Robot é na realidade, uma biblioteca de classes podendo ser usado em combinação com qualquer ambiente de programação Java como: JCreator, JGrasp, BlueJ, Eclipse, etc.

Tomando por base estas classes nativas, o estudante pode criar classes de robôs mais especializadas, com operações em um nível de maior complexidade.

Diversas simulações podem ser planejadas, como:

- Ensinar o robô a efetuar um cálculo (Fig. 12);
- Subir degraus (Fig. 13);
- Corrida com obstáculos (Fig. 14).
- Escapar de labirintos (Fig. 15).

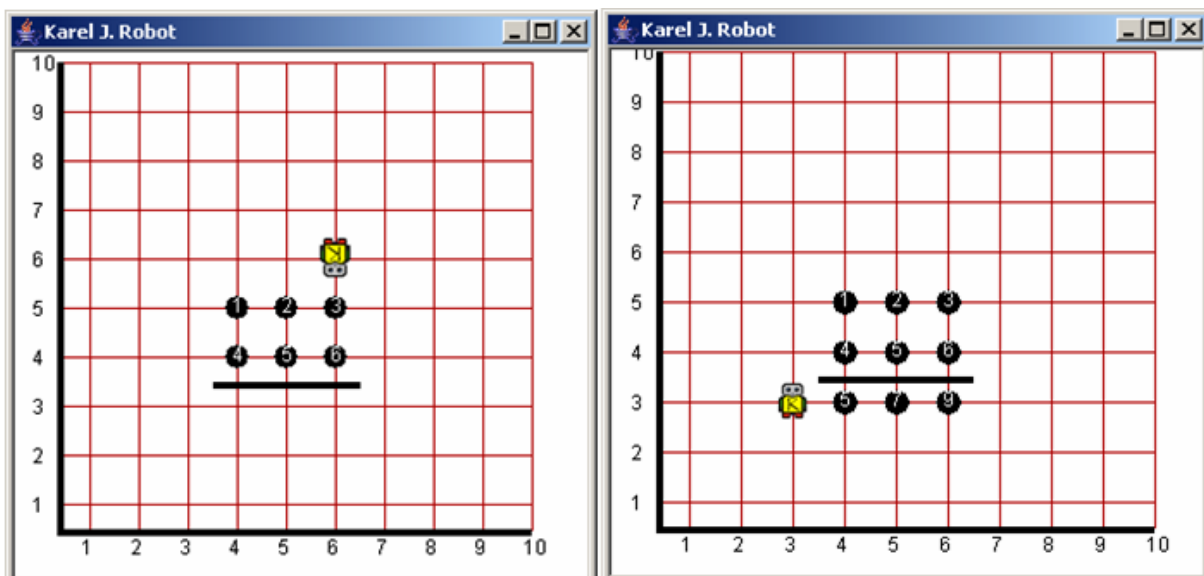


Figura 7 – Cálculo soma situação inicial e final

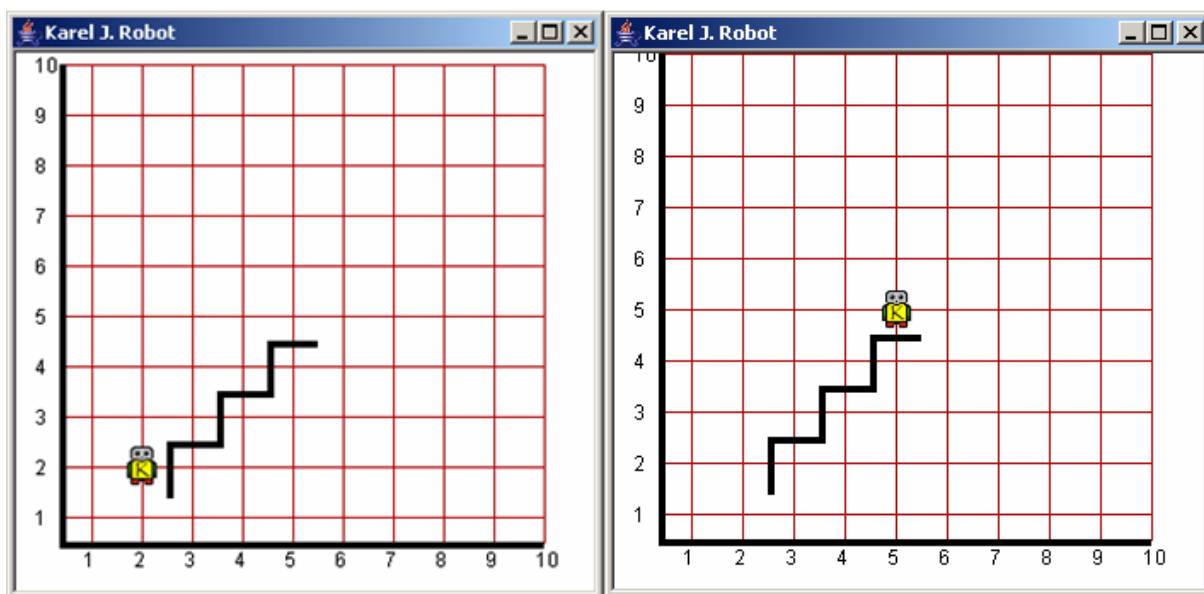


Figura 8 – Subir degraus situação inicial e final

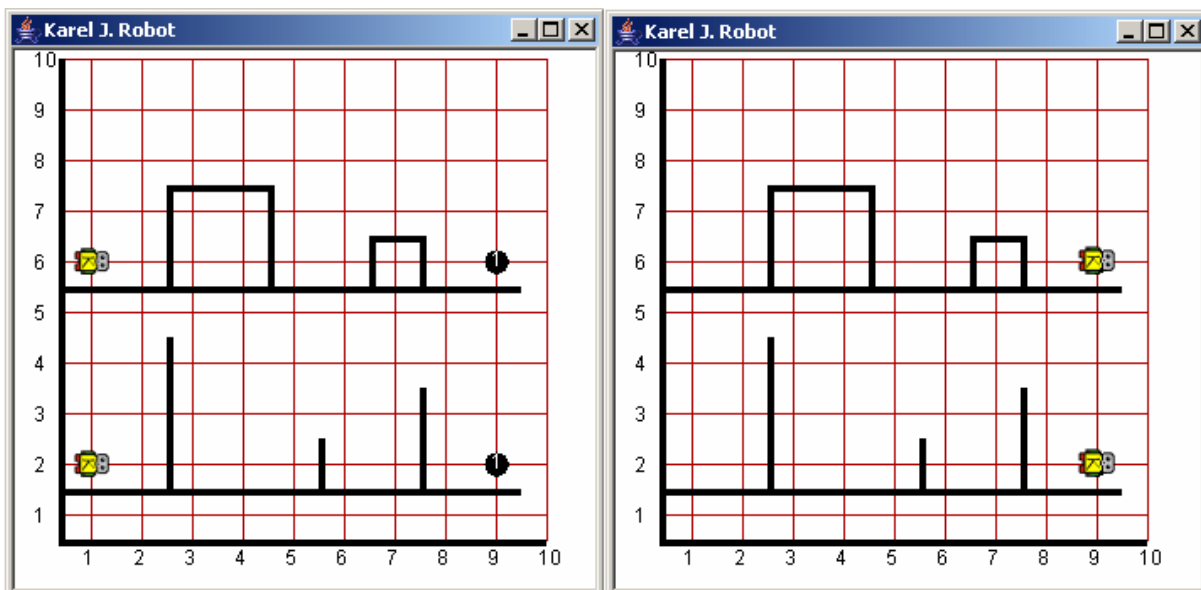


Figura 9 – Corrida com obstáculos situação inicial e final

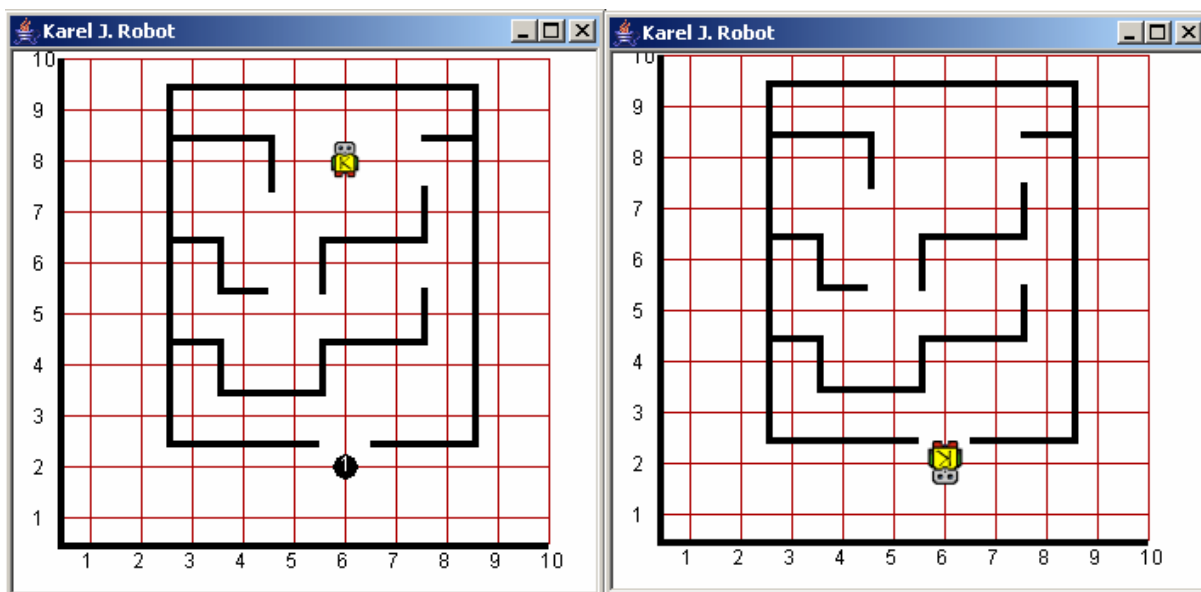


Figura 10 – Escapar de Labirintos: situações inicial e final

O Karel J. Robot pode ser usado no primeiro dia de aula e tem como público alvo estudantes que não têm nenhuma experiência com programação, ou com o paradigma OO.

Analisados os recursos computacionais do software Karel J. Robot, no capítulo que segue, procura-se analisar, sob a ótica de um recurso didático ou meio de ensino, o software Karel J. Robot, de forma a identificar características da mediação pedagógica no uso do mesmo por um professor.

REFERÊNCIAS BIBLIOGRÁFICAS

ADAMS, Joel C.; COLLEGE, Calvin. **Object-centered design**: a five-phase introduction to object-oriented programming in CS1-2. SIGCSE Bulletin, v.28, n.1, p.78-82, Mar. 1996.

ALVAREZ, X.; GONZÁLES, R. M.; PRIETO, M.; ROSSI, G. **Customizing learning environments for teaching OO technology to different Communities**. In: International Conference on Teaching and Training Object-Oriented Technology, Jan. 1995, Londres.

BECK, Kent; CUNNINGHAM, Ward. **A laboratory for teaching object-oriented thinking**. In: Conference on Object-Oriented Programming Systems, Languages, and Applications, Oct. 1989, New Orleans.

BECKER, Karin; ZANELLA, Ana Lucia. **A cooperation model for teaching/learning modeling disciplines**. In: CYTED-RITOS International Workshop on Groupware, 4th, 9-11 Sept., 1998, Búzios (RJ).

BERGIN, Joseph et al. **Karel J. Robot, A Gentle Introduction to the Art of Object Oriented Programming**. New York: Dream Songs Press, 2005.

BORDENAVE, Juan Diaz. PEREIRA, Adair Martins. **Estratégias de ensino-aprendizagem**. 18ª ed. Petrópolis: Editora Vozes, 1998.

DE CLUE, Tim. **Object-orientation and the principles of learning theory**: a new look at problems and benefits. SIGCSE Bulletin, v.28, n.1, p.232 - 236, Mar. 1996.

GOLD Eric; ROSSON Mary Beth. Portia: **An Instanced-Centered Environment for Smalltalk**, OOPSLA 1991, Conference Proceedings on Object-Oriented Programming Systems, languages, and applications, Oct. 6-11, 1991, Phoenix, AZ.

KOLLING, Michael; KOCH, Bett; ROSENBERG, John. **Requirements for a first year object-oriented teaching language**. SIGCSE Bulletin, v.27, n.1, p.173-177, Mar. 1995.

KÖLLING Michael; ROSENBERG John. **Tools and Techniques for Teaching Objects First in a Java Course**. SIGCSE'99 The Proceedings of Thirtieth SIGCSE Technical Symposium on Computer Science Education, March 1999, New Orleans, LA, pp.368.

LTWIN, Edith. Org. **Tecnologia Educacional: Política, histórias e propostas**. Porto Alegre : Artes Médicas, 1997.

MARQUES, Cristina P. C.; MATTOS, Maria Izabel L. de; LA TAILLE, Yves de. **Computador e Ensino: Uma aplicação à língua portuguesa**. São Paulo: Editora Ática, 1986.

MATUI, Jiron. **Construtivismo – Teoria construtivista sócio-histórica aplicada ao**

ensino. São Paulo: Moderna, 1995.

MENDES, A. J.; GOMES, A. **Suporte à aprendizagem da programação com o ambiente SICAS.** In: V Congresso Ibero-Americano de Informática Educativa, 3., 2000, Viña del Mar, Chile.

MORAN, José Manuel; MASETTO, Marcos T.; BEHRENS, Maria Aparecida. **Novas tecnologias e mediação pedagógica.** 3ª ed. Campinas: Papirus, 2000.

MOREIRA, Marco Antonio. **Teorias de aprendizagem.** São Paulo: EPU, 1999.

PFROMM NETTO, Samuel. **Tecnologia da Educação e comunicação de massa.** São Paulo : Pioneira, 1976.

OLIVEIRA, Celina Couto de; COSTA, José Wilson da; MOREIRA, Mercia. **Ambientes Informatizados de aprendizagem:** produção e avaliação de software educativo. Campinas: Papirus, 2001.

OLIVEIRA, Marta Kohl de. **Vigotsky: Aprendizado e desenvolvimento:** Um processo sócio-histórico. São Paulo: Editora Scipioni, 1998.

RUMBAUGH, James. **Modelagem e projetos baseados em objetos.** São Paulo: Campus, 1994.

SANCHO, Juana M. et al. **Para uma tecnologia educacional.** Org.: Juana M. Sancho; Trad. Beatriz Affonso Neves. Porto Alegre: ArtMed, 1998.

SILVA, Cassandra Ribeiro. **MAEP:** Um método ergopedagógico interativo de avaliação para produtos educacionais informatizados. Florianópolis, 2002. Tese (Doutorado em Engenharia de Produção) – Universidade Federal de Santa Catarina.

SOUZA, M. F. C. **Um Ambiente de Apoio à Seleção de Software Educativo** Fortaleza, 2006. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal do Ceará

WIRFS, Rebecca; WILKERSON, Brian; WIENER, Lauren. **Designing object-oriented software.** Englewood Cliffs: Prentice - Hall, 1989.

ZANELLA, Ana Lucia. **Um ambiente colaborativo para apoio a um curso de projeto de software orientado a objetos** Porto Alegre, 1999. Dissertação (Mestrado em Informática) – Pontifícia Universidade Católica do Rio Grande do Sul.