

Ensino de lógica de programação baseado na indução-dedução através de exemplos

Leandro H. Nagano¹, Alexandre I. Direne¹

¹Programa de Pós-Graduação em Informática
Universidade Federal do Paraná - UFPR
Curitiba - PR - Brasil

{lhnagano, alexd}@inf.ufpr.br

Abstract. *The difficulty encountered in programming logic discipline is one of the reasons for the avoidance in computer courses. To assist in the learning of this discipline, an intelligent tutor system can be designed addressing pedagogical and computational concepts. In this article, will be presented the design and implementation of a teaching method based on proposals Felder-Silverman and Vigotsky, referring to examples based learning.*

Resumo. *A dificuldade encontrada na disciplina de lógica de programação é um dos motivos para a evasão nos cursos de computação. Para auxiliar no aprendizado dessa disciplina, um sistema tutor inteligente pode ser projetado abordando conceitos pedagógicos e computacionais. Nesse artigo, será apresentada a concepção e implementação de um método de ensino baseado nas propostas de Felder-Silverman e Vigotsky, que remetem a aprendizagem baseada em exemplos.*

1. Introdução

A disciplina de lógica de programação, segundo Rocha et al (2010, p.3), é “requisito fundamental nos cursos de computação e um dos grandes motivos de evasão escolar nos cursos de computação”, fatores que justificam sua escolha como objeto de estudo.

De acordo com o Parecer CNE/CES nº 136/2012, nos cursos de graduação em computação, "a metodologia de ensino deve ser centrada no aluno como sujeito da aprendizagem e apoiada no professor como facilitador do processo de ensino-aprendizagem".

Um sistema tutor inteligente é projetado abordando conceitos pedagógicos e computacionais, auxiliando o processo de ensino e amenizando as dificuldades encontradas pelo aprendiz. Um conceito pedagógico que pode ser citado é a proposta de aprendizagem.

Nesse artigo, as seguintes propostas de aprendizagem serão abordadas:

Felder-Silverman que “classifica os aprendizes em cinco dimensões: ativos-reflexivos; sensoriais-intuitivos; visuais-verbais; indutivos-dedutivos; sequenciais-globais”.

Vigotsky, que refere-se aos processos mentais que estão em construção através do conceito de zona de desenvolvimento proximal e do bom ensino, que aproveita o potencial do aluno.

Essas duas propostas podem contribuir para o ensino de lógica de programação através do sistema tutor inteligente, desenvolvendo o potencial do aluno através da dimensão indutiva-dedutiva e da aprendizagem baseada em exemplos.

2. Metodologia

A metodologia empregada nesse artigo foi a pesquisa bibliográfica, através da pesquisa em revistas, teses, dissertações e anais de eventos científicos relacionados ao tema de ensino de lógica de programação.

3. Fundamentação Teórica

A Informática na Educação oferece os recursos que podem auxiliar no processo de ensino personalizado e individualizado através de ambientes de aprendizagem, que possibilitam ao aluno estudar segundo seu ritmo e seus objetivos, potencializando a aprendizagem de um conteúdo. (Falckembach; Araujo, p. 2). Um desses recursos é o sistema tutor inteligente, que é um ambiente de aprendizagem que auxilia no processo de ensino.

Acredita-se que o ensino de algoritmos deve ser baseado na programação tradicional, apresentando aos alunos os procedimentos básicos de forma gradual a fim de levar o aluno a confiar em suas potencialidades. (Falckembach; Araujo, p. 2). Os conceitos pedagógicos encontrados nas propostas de aprendizagem de Felder-Silverman e de Vigotsky tem o objetivo de auxiliar o processo de ensino do aprendiz, baseados na dimensão indutiva-dedutiva e na teoria do bom ensino.

As formas como os estudantes organizam a informação, segundo Felder, pode ser indutiva ou dedutiva. Os indutivos preferem partir de casos específicos e ir construindo até chegar aos princípios e teorias fundamentais por inferência; já os estudantes dedutivos preferem começar com princípios e regras gerais e então deduzir as conseqüências e as aplicações. (Senra et al, p. 3)

A zona de desenvolvimento proximal refere-se aos processos mentais que estão em construção. O bom ensino, na perspectiva vygotskiana, seria aquele que aproveita o potencial do aluno e dá apenas o “empurrão para que esse possa se desenvolver. (Cury, p. 5-6)

Para a aplicação da dimensão indutiva-dedutiva de Felder-Silverman e da teoria do bom ensino de Vigotsky, podemos citar a aprendizagem baseada em exemplos. A maneira mais difundida para se ensinar a programação é através do estudo de exemplos prontos e da aplicação do algoritmo destes exemplos em problemas semelhantes.

A solução independente de problemas, seja pela dimensão indutiva ou dedutiva, possibilita o aprendiz a atingir a zona de desenvolvimento proximal. O incentivo para que o potencial do aluno possa ser desenvolvido pode ser um exemplo ou uma visão geral da teoria.

Com a aplicação do mesmo algoritmo em um ou mais problemas semelhantes o aluno adquire efetivamente o conhecimento embutido no conjunto de comandos que formam este algoritmo. O aluno novato é induzido a resolver certo conjunto de problemas de uma determinada maneira e assim adquire a perícia ou a capacidade de agrupar comandos para formar soluções através da repetição do uso de um conjunto de comandos que foram apresentados através de um exemplo pronto. (Binder, p. 5)

4. Solução Proposta

Quando o aprendiz acessa o sistema tutor, os exercícios de lógica de programação são disponibilizados para que possam ser resolvidos. O sistema não tem como objetivo reconhecer o aluno de acordo com os perfis de comportamento citados por Felder-Silverman.

Quando o aprendiz seleciona um exercício, é apresentado o tópico abordado pelo exercício, as opções “Ver exemplo” e “Visão geral da teoria” e o enunciado do exercício proposto.

A figura 1 demonstra um exercício que se refere ao tópico estruturas de condição.

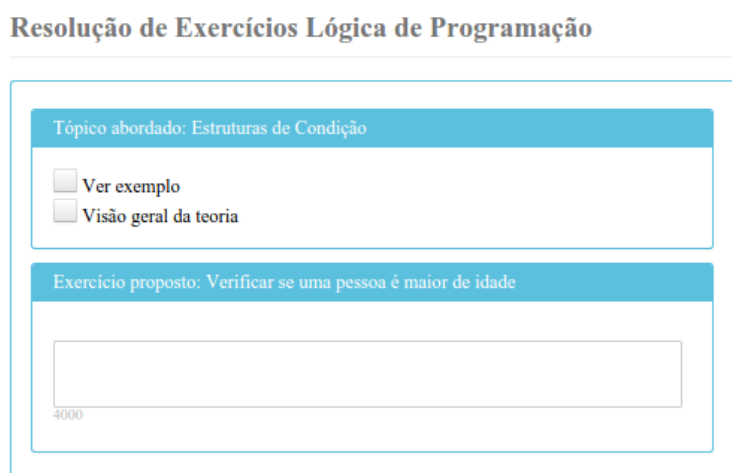


Figura 1. Exercício do tópico estruturas de condição

Se o aprendiz encontrar dificuldades na resolução do exercício, pode clicar em “Ver exemplo”, para visualizar um exemplo do tópico abordado ou em “Visão geral da teoria”, para visualizar a visão geral da teoria do tópico abordado.

A figura 2 demonstra um exemplo do tópico estruturas de condição.

Resolução de Exercícios Lógica de Programação

Tópico abordado: Estruturas de Condição

Ver exemplo
 Visão geral da teoria

Exemplo: Ler dois números e determinar o maior

```
n1, n2: inteiro;  
início  
  escreva("Informe o primeiro número: ");  
  leia(n1);  
  escreva("Informe o segundo número: ");  
  leia(n2);  
  se (n1 > n2)  
  então  
    escreva("O maior número é: ", n1);  
  senão  
    escreva("O maior número é: ", n2);  
fim-se  
fim
```

3739

Exercício proposto: Verificar se uma pessoa é maior de idade

4000

Figura 2. Exemplo do tópico abordado

A figura 3 demonstra a visão geral da teoria do tópico estruturas de condição.

Resolução de Exercícios Lógica de Programação

Tópico abordado: Estruturas de Condição

Ver exemplo
 Visão geral da teoria

Visão Geral da Teoria

Muda o fluxo das instruções de um algoritmo de acordo com uma condição

Estrutura básica:

```
se <condição>  
  então  
    <Instruções a serem executadas caso a condição seja verdadeira>  
  senão  
    <Instruções a serem executadas caso a condição seja falsa>  
fim-se
```

3832

Exercício proposto: Verificar se uma pessoa é maior de idade

4000

Figura 3. Visão geral do tópico abordado

5. Conclusão

Nesse artigo foi proposto um sistema tutor inteligente baseado na aprendizagem baseada em exemplos. A dimensão indutiva-dedutiva de Felder-Silverman e a teoria do bom ensino de Vigotsky mostram-se adequadas para esse sistema, pois o sistema tutor demonstra um exemplo e uma visão geral da teoria para o tópico abordado pelo exercício, desenvolvendo processos mentais, potencializando o entendimento e o aprendizado.

Referências

- Binder, F. V. (1999) "Conceitos e Ferramentas para Apoiar o Ensino de Lógica de Programação Imperativa". In: X Simpósio Brasileiro de Informática na Educação.
- Cury, H. N. (2000) "Estilos de Aprendizagem de Alunos de Engenharia". In: XXVIII Congresso Brasileiro de Ensino de Engenharia.
- Falckembach, G. A. M.; Araujo, F. V. (2006) "Aprendizagem de Algoritmos: Dificuldades na Resolução de Problemas", In: Congresso Sul Brasileiro de Computação.
- Rocha, P. S.; Ferreira, B.; Monteiro, D.; Nunes, D. S. C.; Góes, H. C. N. (2010) "Ensino e Aprendizagem de Programação: Análise da Aplicação de Proposta Metodológica Baseada no Sistema Personalizado de Ensino" In: Novas Tecnologias na Educação
- Senra, C. M. S.; Lima, G. F. C. A.; Silva, F. W. O. (2014) "A Relação entre os Estilos de Aprendizagem de Richard Felder e os Tipos Psicológicos de Carl Jung". In: IV Seminário Nacional de Educação Profissional e Tecnológica.