

The background is a light blue sky with a large yellow sun in the top right corner. There are three white, fluffy clouds. The bottom of the image shows a green rolling landscape with two stylized green trees on the left and two on the right, with small pink flowers scattered on the grass.

Pensamento Computacional

https://github.com/rodrigoeifr/oa_pensamentocomp1

Rodrigo Elias Francisco

Palestrante

- Professor efetivo do IF Goiano desde 2012.
- Mestre em Ciência da Computação pela UFG.
Cursando Doutorado na mesma área pela UFU.
- Bacharel em Sistemas de Informação e
Especialista em Segurança da Informação.
Trabalhou como Analista de Sistemas por 8 anos.
- Realiza pesquisas sobre Inteligência Artificial e
Engenharia de Software aplicados na Educação



Algumas Publicações

iSys - Brazilian Journal of Information Systems

[Sobre](#) ▾ [Atual](#) [Arquivos](#) [Submissões](#) [Notícias](#) [Contato](#)

[Início](#) / [Arquivos](#) / [v. 9 n. 2 \(2016\)](#) / ARTIGOS DE EDIÇÃO ESPECIAL

Uso do Algoritmo Distância de Edição com Técnicas de Pré-Processamento para Apoiar a Identificação de Plágio em Códigos-Fonte de Problemas de Programação Introdutória

Rodrigo Elias Francisco

IFGoiano

Ana Paula Laboissière Ambrósio

UFG

 PDF

Publicado

2016-07-27

Resumo

Este trabalho aborda o problema de plágio no ensino de programação introdutória no contexto de um sistema de administração e correção automática de listas de exercícios. Com o objetivo de identificar uma ferramenta de detecção de plágio que pudesse ser incorporada no sistema

Como Citar

Francisco, R. E., & Ambrósio, A. P. L. (2016).
Uso do Algoritmo Distância de Edição com
Técnicas de Pré-Processamento para Apoiar a
Identificação de Plágio em Códigos-Fonte de
Problemas de Programação Introdutória. *ISys*



Algumas Publicações

REVISTA BRASILEIRA DE INFORMÁTICA NA EDUCAÇÃO (BRAZILIAN JOURNAL OF COMPUTERS IN EDUCATION)

[CAPA](#) [SOBRE](#) [ACESSO](#) [CADASTRO](#) [PESQUISA](#) [ATUAL](#) [ANTERIORES](#) [INSTRUÇÕES AOS AUTORES](#) [EDIÇÕES ESPECIAIS](#)

[Capa > v. 26, n. 03 \(2018\) > Francisco](#)

Juiz Online no ensino de CS1 - lições aprendidas e proposta de uma ferramenta

Rodrigo Elias Francisco, Ana Paula Laboissière Ambrósio, Cleon Xavier Pereira Junior, Márcia Aparecida Fernandes

Resumo

Os juizes online, inicialmente usados em maratonas de programação, vêm sendo adotados também para o ensino de Programação Introdutória (CS1), apresentando algumas vantagens, como a redução da carga de trabalho do professor e o feedback instantâneo ao estudante, e desvantagens por não serem totalmente adaptados como ferramenta para o ensino. Este trabalho apresenta uma Revisão Sistemática da Literatura (RSL) acerca de juizes online para o ensino de CS1, que se baseou nos seguintes aspectos: benefícios, problemas; requisitos funcionais e não-funcionais. O resultado da revisão contribuiu para especificar um juiz online com foco no suporte à disciplina de CS1. Também foi realizada uma experiência prática com o uso do juiz online BOCA (desenvolvido para maratonas de programação) no ensino de turmas matriculadas em CS1. As lições aprendidas na experiência prática e os conhecimentos adquiridos na RSL contribuíram para uma proposta de juiz online para o ensino de CS1, com foco principalmente em três requisitos considerados essenciais: construção de listas de exercícios, feedback personalizado, e plágio. Assim, esta pesquisa contribui com o ensino de programação introdutória ao apresentar uma abordagem baseada em resultados encontrados na literatura, através de artigos que apresentam diferentes juizes online para o ensino, e experiências práticas com turmas reais.

Palavras-chave

juiz online; CS1; Programação Introdutória; RSL; Feedback; Plágio

Algumas Publicações

SCITEPRESS
DIGITAL LIBRARY

Research.Publish.Connect.

Search by any title, abstract, authors

A Recommendation Module based on Reinforcement Learning to an Intelligent Tutoring System for Software Maintenance

Topics: Architectures for AI-based Educational Systems; Intelligent Tutoring Systems

In Proceedings of the 14th International Conference on Computer Supported Education - Volume 1: CSEDU, 322-329, 2022

A Recommendation Module based on Reinforcement Learning to an Intelligent Tutoring System for Software Maintenance

Rodrigo Elias Francisco^{1,2} and Flávio de Oliveira Silva¹

¹Faculty of Computer, Federal University of Uberlândia (UFU),

Av. João Naves de Ávila, 2121, Block 1A, Room 1A243 - Campus Santa Mônica, Uberlândia, MG, Brazil

²Federal Institute Goiano (IF Goiano) - Campus Morrinhos, Rodovia BR153, KM633 Zona Rural, Morrinhos, GO, Brazil

Keywords: Intelligent Tutoring System, Software Maintenance, Reinforcement Learning, Q-Learning.

Abstract: The demand for qualified professionals to work with Software Maintenance (SM) brings challenges to computer education. These challenges are related to SM's inherent complexity and the teacher's significant work in providing adequate support in practical SM activities. In this context, Artificial Intelligence (AI) based techniques, such as recommendations, can play a central role in developing Intelligent Tutoring Systems (ITS) to focus the teaching-learning process. The literature points out a lack of ITS to SM and that most of them do not use AI-based techniques to recommend content to the students. In this work, we present an Expert Knowledge Module (EKM) for an ITS specially designed for SM. To model the EKM content, we did a deep analysis of the ACM curricula regarding SM topics and the syllabus related to SM from all Brazilian public universities. The content recommendation engine uses the Q-Learning algorithm, a well-known Reinforcement Learning (RL) AI-based technique. Using simulation-based experiments, we could verify the efficiency of the Q-Learning-based recommendation mechanism to propose contents using the ITS's EKM property. This work highlights how AI-based techniques can enhance and improve SM teaching-learning process using ITS and advance this research area.

1 INTRODUCTION

Software maintenance (SM) is responsible for about 60% of all costs (Russell and Vinel, 2017) during the software life-cycle (Russell and Vinel, 2017) due to high consumption of time and effort (Fernandez-Sia et al., 2014). Learning SM involves handling complex tasks related to the SM process, understanding and modifying software artifacts. Therefore, computer education needs to help students become professionals capable of working with SM (Bleckman et al., 2018).

This educational problem brings motivations to design educational tools based on AI for the SM teaching-learning process, such as Intelligent Tutoring Systems (ITS). ITS is a software system to enhance, adapt, and automate (Alkhawariz and Rafiq, 2014) the teaching-learning process. Although there is ITS's with different architectures, the ITS generally works with representations of the three types of knowledge: the content, the student, and teaching

SCIENCE AND TECHNOLOGY PUBLICATIONS

strategies. The literature about ITS for SM presents some gaps. Few works on tutor systems for SM and fewer on ITS to SM. Most of the work in the literature does not detail content recommendations. None of the ones present some recommendation mechanism that uses AI-based techniques.

In this work we present the design of the Expert Knowledge Module (EKM) Expert Knowledge Module (EKM) module of a ITS for SM and a content recommendation mechanism based on the Q-Learning algorithm, a well-known Reinforcement Learning (RL) AI-based technique used in several areas (Sawsky and Balady, 2018). The main contributions of this work are:

- An ITS with an EKM can handle different types of content related to SM;
- The EKM modeling resulted from a comprehensive analysis of the computer science curriculum proposed by scientific societies, namely the Brazilian Computer Society (SBC) (Zorzo et al., 2017), and the Association for Computing Machinery (ACM) (ACM Computing Curricula Task

Authors: Rodrigo Francisco^{1,2} and Flávio Silva¹

Affiliations: ¹ Faculty of Computer, Federal University of Uberlândia (UFU), Av. João Naves de Ávila, 2121, Block 1A, Room 1A243 - Campus Santa Mônica, Uberlândia, MG, Brazil ; ² Federal Institute Goiano (IF Goiano) - Campus Morrinhos, Rodovia BR153, KM633 Zona Rural, Morrinhos, GO, Brazil

Keyword(s): Intelligent Tutoring System, Software Maintenance, Reinforcement Learning, Q-Learning.

Abstract: The demand for qualified professionals to work with Software Maintenance (SM) brings challenges to computer education. These challenges are related to SM's inherent complexity and the teacher's significant work in providing adequate support in practical SM activities. In this context, Artificial Intelligence (AI) based techniques, such as recommendations, can play a central role in developing Intelligent Tutoring Systems (ITS) to focus the teaching-learning process. The literature points out a lack of ITS to SM and that most of them do not use AI-based techniques to recommend content to the students. In this work, we present an Expert Knowledge Module (EKM) for an ITS specially designed for SM. To model the EKM content, we did a deep analysis of the ACM curricula regarding SM topics and the syllabus related to SM from all Brazilian public universities. The content recommendation engine uses the Q-Learning algorithm, a well-known Reinforcement Learning (RL) AI-based technique. Using s (More)



Pensamento Computacional

- “A combinação do **pensamento crítico** com os fundamentos da **computação** define uma metodologia para **resolver problemas**, denominada **pensamento computacional (PC)** [Wing 2006]. (Andrade et al., 2013)”
- “PC é um método para **solução de problemas** baseado nos fundamentos e técnicas da **Ciência da Computação** [Wing 2006]. (Andrade et al., 2013)”



Pensamento Computacional

- “Na computação, *hashing* constitui-se de um método intuitivo, que permite organizar grandes quantidades de dados e possui como ideia central a divisão de um universo de dados em subconjuntos mais gerenciáveis.

No cotidiano, o processo de **organizar porcas e parafusos em caixas** e seções apropriadas também constitui uma forma de *hashing*.

(Andrade et al., 2013)”



Pensamento Computacional

- “Conforme [Wing 2006], o PC pode ser colocado como uma das **habilidades intelectuais básicas** de um ser humano, comparada a **ler, escrever, falar e fazer operações aritméticas**. Habilidades estas que servem para descrever e explicar situações complexas. (Andrade et al., 2013)”



Pensamento Computacional

- “PC é um processo de solução de problemas que inclui:
 - i) Formular problemas visando solução via computador
 - ii) Organizar os dados logicamente visando análise
 - iii) Representar dados através de abstrações (modelos e simulações)
 - iv) Automatizar as soluções através de algoritmos
 - v) Identificar, analisar e implementar soluções possíveis visando eficiência e eficácia
 - vi) Generalizar a solução para aplicar a outros problemas.
- Adaptado de (Andrade et al., 2013)”



Pensamento Computacional

- “Três pilares básicos que fundamentam o PC: **abstração, automação e análise.**
- A **abstração** é a capacidade de extrair apenas as características importantes de um problema. Ajuda a lidar com a complexidade.
- A **Automação** é o uso de um meio eletrônico na substituição do trabalho manual. Um computador é um exemplo.
- A **análise** é o estudo dos resultados gerados pela automação.
- Adaptado de (Andrade et al., 2013)”



Pensamento Computacional

- **“Computational Thinking in K-12 Education Leadership Toolkit**

Traz nove conceitos da área da Computação fundamentais para o desenvolvimento do PC

Adaptado de (Andrade et al., 2013)”



Pensamento Computacional

- **“Computational Thinking in K-12 Education Leadership Toolkit**
 - 1. Coleta de dados: reunir dados de forma apropriada
 - 2. Análise de dados: tornar os dados coletados coerentes, encontrar padrões e tirar conclusões a partir destes dados
 - 3. Representação de dados: organizar informações por meio de tabelas, gráficos, palavras, imagens ou outro recurso
 - 4. Decomposição de problemas: divisão das tarefas em partes menores e manuseáveis.

Adaptado de (Andrade et al., 2013)”



Pensamento Computacional

- **“Computational Thinking in K-12 Education Leadership Toolkit**
 - 5. Abstração: redução da complexidade de um problema para focar na questão principal
 - 6. Algoritmos e procedimentos: série organizada de passos para atingir algum objetivo.
 - 7. Automação: utilização de computadores ou máquinas para fazer tarefas repetitivas ou tediosas
 - 8. Simulação: representação ou a modelagem de um processo e a sua execução.
 - 9. Paralelismo: organizar recursos para simultaneamente desenvolver tarefas que atinjam um objetivo em comum

Adaptado de (Andrade et al., 2013)”



Pensamento Computacional

- Como o PC pode ajudar na aprendizagem de Programação?



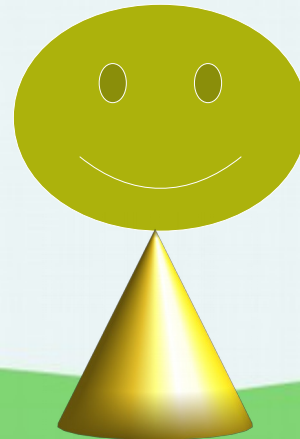
Pensamento Computacional

- Como o PC pode ajudar na aprendizagem de Programação?
 - Usar as 9 fases para a resolução de problemas

1. Coleta de dados
2. Análise de dados
3. Representação de dados
4. Decomposição de problemas
5. Abstração
6. Algoritmos e procedimentos
7. Automação
8. Simulação
9. Paralelismo



Insight: utilizar o meu repertório de Padrões



Minha experiência como Professor

- PC requer paciência e resolução de MUITOS problemas
- A prática de programação ajuda a desenvolver intuitivamente PC
- As habilidades de PC vão ajudar em matérias distintas, como:
 - Modelagem UML
 - Orientação a Objetos
 - Inteligência Artificial
 - Análise de Algoritmos
 - Linguagem de Programação
 - **RESOLUÇÃO DE PROBLEMAS!**
- O desenvolvimento de PC precisa de um ciclo (ler, resolver, obter feedback) que ficou prejudicado na **PANDEMIA**



Vamos treinar nossas habilidades de PC

- Problema 1): (10 minutos / discuta em grupos e anotem as etapas)
 - Suponha que você seja um fazendeiro(a) que possua 200 vacas leiteiras.
 - Gere aleatoriamente um número de 30 a 100 que irá representar a produção mensal de cada vaca.
 - Faça um programa que responda:
 - Qual a menor/menor quantidade de leite produzida e qual vaca foi a responsável
 - Quantos litros de leite foram produzidos no total
 - A lista de vacas/produção ordenadas por produção
- a) Implemente como (procedural)
- b) Implemente como (orientado a objetos)

1. Coleta de dados
2. Análise de dados
3. Representação de dados
4. Decomposição de problemas
5. Abstração
6. Algoritmos e procedimentos
7. Automação
8. Simulação
9. Paralelismo



Vamos treinar nossas habilidades de PC

- Problema 2): (10 minutos / discuta em grupos e anotem as etapas)
 - Modifique a solução anterior
 - Gere aleatoriamente um número de 0 a 100 que irá representar a produção de cada vaca.
 - O programa anterior deve agora incluir:
 - A lista de vacas com produção extremamente baixa (0 a 30 litros mensais)
 - A opção para o usuário escolher qual funcionalidade executar
- A) Implemente como (orientado a objetos)

1. Coleta de dados
2. Análise de dados
3. Representação de dados
4. Decomposição de problemas
5. Abstração
6. Algoritmos e procedimentos
7. Automação
8. Simulação
9. Paralelismo



Quero um feedback dos alunos

- Como usaram o PC nas 2 questões?
- Como treinar as habilidades de PC para evoluírem como estudantes?
- Como relacionar a Abstração com a Orientação a Objetos?

1. Coleta de dados
2. Análise de dados
3. Representação de dados
4. Decomposição de problemas
5. Abstração
6. Algoritmos e procedimentos
7. Automação
8. Simulação
9. Paralelismo



Quero um feedback dos alunos

- Perguntas?
- Obrigado!!

1. Coleta de dados
2. Análise de dados
3. Representação de dados
4. Decomposição de problemas
5. Abstração
6. Algoritmos e procedimentos
7. Automação
8. Simulação
9. Paralelismo



Minhas pesquisas no IF

- Sistemas Tutores Inteligentes para Educação em Computação
 - Ler e compreender código-fonte e artefatos de software
 - Recomendação de conteúdo educacional



Minhas pesquisas no IF

Sistema Tutor Inteligente Engenharia de Software Educacional + IA na Educação

Pensam. Comput.

Compreensão
de Software

código-fonte

pergunta

Conhecimento Edu:

- Taxonomia de bloom
- Metacognição
- Pens. Computacional

Dicas Metacognitivas
com IA

Posso te
ajudar?

Modelo do
Estudante.
Como represent.
no
computador?

Recomendação de Conteúdo com
Intelig. Artificial (IA)

Exercícios a fazer: x, y, a, b.... z

Conversando sobre o
programa que você,
estudante, está lendo
/compreendendo..

Tem dúvida

Diálogo direcionado
por IA

chatbot



Referências

- Andrade, Daiane, et al. "Proposta de atividades para o desenvolvimento do pensamento computacional no ensino fundamental." Anais do XIX Workshop de Informática na Escola. SBC, 2013.

