

O ensino de física no ensino médio intermediado por programação em linguagem Python

Teaching Physics in high school through Python programming language

Rodrigo Amarante Colpo

Universidade do Estado do Rio de Janeiro - Instituto de Física Armando Dias Tavares
rodrigocolpo2@yahoo.com.br

Artur Uhlig de Faria

Universidade do Estado do Rio de Janeiro - Instituto de Física Armando Dias Tavares
arturfaria.fisica@gmail.com

Alan Freitas Machado

Universidade do Estado do Rio de Janeiro - Instituto de Física Armando Dias Tavares
alanfmac@gmail.com

Resumo

Este trabalho tem como objetivo descrever um projeto em que foi utilizada programação em linguagem Python como suporte para o ensino de Física no ensino médio. Testamos metodologias de ensino junto a alunos do 3º ano do ensino médio em uma escola pública do Rio de Janeiro, em formato de oficina, priorizando o desenvolvimento do pensamento computacional através de diálogos em que os alunos foram estimulados e construir coletivamente conhecimentos.

Palavras-chave: física, ensino de física, educação, programação, python

Abstract

This paper has the objective of describing a Physics teaching project that used Python language programming to support Physics teaching in high school. We tested teaching methods along with third grade high school students from a public school in Rio de Janeiro, in the form of practical classes, aiming the development of computational thinking through dialogues with the said students in which they were stimulated to collectively build knowledge.

Keywords: physics, physics teaching, education, programming, python

Introdução

O *Programa Internacional de Avaliação de Alunos (PISA)* realiza a aplicação de provas mundialmente desde 2000, a cada 3 anos, com o objetivo de avaliar a qualidade de ensino de matemática, ciências e leitura de alunos com 15 anos, tendo sido realizado, em sua última edição, em 65 países. Com relação ao período entre 2009 (OECD, 2010) e 2012 (OECD, 2013), o Brasil teve uma melhora de 1,27% em sua nota na prova de matemática e 0% em sua nota em ciências, enquanto os 20 últimos qualificados que participaram das duas provas tiveram uma melhora de 1,54% em suas notas em matemática e 0,84% em ciências.

Os resultados do PISA apontam para um quadro problemático da educação brasileira e mobiliza o governo a propor soluções. Dentre as soluções que visam a melhoria do ensino, está a modernização das escolas e das salas de aula, com projetos de inclusão digital, como o ProInfo, que entre 1997 e 2010 distribuiu mais de 30 mil laboratórios de informática (CARRANO, 2010), e o programa Um Computador Por Aluno, que oferece laptops para que alunos possam utilizá-los dentro de sala de aula (CAPPELLETTI, 2012).

Para a instalação dos laboratórios de informática, o ProInfo exige que as escolas tenham mais de 30 alunos e tenham energia elétrica (Proinfo, 2015), e não faz qualquer referência a um projeto pedagógico para utilização desses espaços. Como resultado, observa-se que o simples acréscimo de equipamentos de informática traz pouco impacto na vida acadêmica dos alunos, e pode ter inclusive impacto negativo (FIRPO, 2012), com muitos se tornando centros de recreação, e não de estudo.

A adequada utilização dos laboratórios de informática é restringida principalmente por três fatores: disponibilização de computadores em boas condições, o que depende de financiamento estadual ou federal; a existência de planos pedagógicos; e a qualificação dos professores responsáveis pela utilização de tais tecnologias de ensino. Nesse contexto, em uma parceria entre a UERJ e o Colégio Pedro II - Regional Centro, criamos um projeto piloto para desenvolver metodologias pedagógicas para o ensino de Física, utilizando a linguagem de programação Python como ferramenta de aprendizagem, com uma turma de 6 alunos de iniciação científica júnior do projeto jovens talentos (FAPERJ-CECIERJ).

Metodologia

O pensamento computacional foi descrito por Wing (2006) como uma forma de abordar problemas a partir da sua divisão em problemas menores e de mais fácil trato, criando um algoritmo mental para a sua solução. Essa é uma habilidade fundamental a todo indivíduo e que pode ser estimulada com recursos computacionais, onde é necessária a sistematização e detalhamento do processo de obtenção dos resultados.

O processo de aprendizado dos conceitos iniciais das linguagens de programação é complexo e marcado pela presença de inúmeras dificuldades, que são detectadas em pesquisas no mundo todo (JÚNIOR, PEREIRA e RAPKEJEWICZ, 2004). Alguns dos pontos mais destacados são a baixa capacidade em resolução de problemas aliadas a equívocos na formulação de modelos mentais adequados, a falta de motivação para executar tarefas, a dificuldade de abstração e linguagens não adaptadas para fins pedagógicos.

Para contornar a utilização de sintaxes complexas, que necessitam de muito tempo para a correção de erros no código, e que podem levar a uma diminuição da autoconfiança e motivação do aluno (Wen, 2014), escolhemos utilizar programação em linguagem Python, que tem sintaxes curtas e simples, com codificação intuitiva. Python também tem as vantagens adicionais de ser gratuito, orientado a objetos e aceitar a criação de funções, permitindo introduzir grande número de exercícios de Física, sem que o foco principal do curso seja na programação.

O curso teve duração de 6 aulas, com 1 hora de duração cada, entre outubro e dezembro. Nestas, foram abordados conceitos de programação baseados na linguagem Python (sintaxes, estrutura de codificação, etc.), com realização de exercícios de fixação baseados em problemas do conteúdo de física do ensino médio.

A. Objetivo das aulas:

As aulas tinham por objetivo utilizar-se de Python para ensinar física, provocando a reestruturação do conhecimento com a aplicação de diferentes abordagens computacionais para resolver problemas físicos, tendo sido abordadas questões de cinemática.

B. Ementa da oficina:

Aula 1: Introdução à programação:

- 1.1. Conceito de algoritmo;
- 1.2. “Programação” com pseudocódigo;
- 1.3. Introdução á sintaxe do Python.

Aula 2: Códigos para MRU:

- 2.1. Revisão de Movimento Retilíneo Uniforme - MRU;
- 2.2. Criação de códigos simples para resolver a equação do MRU utilizando “if”, “for” e “while”.

Aula 3: Códigos para MRUA:

3.1. Revisão da equação de “Torricelli” para Movimento Retilíneo Uniformemente Acelerado - MRUA;

- 3.2. Reestruturação dos códigos anteriormente citados, embasados em diferentes conceitos físicos (Equação de Torricelli, segunda lei de Newton, equação das energias cinética, potencial gravitacional e potencial elástica).

Aula 4: Variáveis de armazenamento:

- 4.1. Listas;
- 4.2. Tuplas;
- 4.3. Exercícios envolvendo os conceitos de “lista” e “tupla” sobre Movimento Retilíneo Uniforme (Armazenamento e manipulação de dados).

Aula 5: Matrizes:

- 5.1. Conceituação de matrizes em linguagem computacional;
- 5.2. Operações matemáticas com matrizes;

5.2. Exercícios sobre função horária da posição, velocidade e aceleração dispostas em forma matricial.

Aula 6: Continuação de Matrizes:

6.1. Exercícios com matrizes, utilizando-as para gravar dados de diferentes simulações (fazendo alteração de posição inicial, velocidade inicial, etc.) e posteriormente compará-los.

C. Avaliação:

Os alunos foram avaliados de modo contínuo durante as aulas, enquanto eram estimulados a criar algoritmos e a “pensar em voz alta” acerca das etapas para a resolução dos problemas. A utilização de programação faz com que os erros dos alunos costumem ser principalmente de duas categorias: de sintaxe e de conteúdo. Os próprios alunos percebem rapidamente se houve algum problema de sintaxe, quando o programa não “roda”. Dificuldades conceituais são esclarecidas com o recurso da escuta ativa às dúvidas dos alunos, e são também evidenciadas ao revisitar um mesmo problema físico, segundo outra abordagem computacional, utilizando recursos recém-estudados.

A escolha dos conteúdos computacionais na ementa da oficina teve como base oferecer aos alunos o conhecimento necessário para que criassem códigos que possam resolver diversos problemas com relativamente pequena bagagem de programação. Ao aprender novos conceitos, como por exemplo matrizes, problemas já vistos nas aulas anteriores eram revisitados e novos algoritmos eram criados utilizando-se outros recursos, permitindo visualizar as mesmas questões sob outra perspectiva, ampliando a compreensão da física abordada e desenvolvendo o pensamento computacional dos alunos. Essa etapa foi particularmente importante para que o professor recebesse *feedbacks* das dificuldades dos alunos.

O conteúdo de física foi abordado utilizando a proposição de problemas e a construção coletiva de soluções, elaborados em nível crescente de dificuldade, de modo a reforçar a autoconfiança e motivação do aluno. A resolução dos problemas foi conduzida de modo a construir coletivamente modelos que os explicassem, sendo os alunos estimulados a “pensar em voz alta” sobre as etapas necessárias para a sua resolução.

O propósito do professor enquanto educador é o de instigar o aluno a transferir o conhecimento previamente adquirido de situações pertinentes ao dia-a-dia para o de situações escolares (PAIS, 2008, p.53). Para se alcançar tal resultado, o emprego de situações-problema é administrado. Ao se deparar com uma situação-problema imposta pelo professor, o aluno consegue transpor a barreira entre o conhecimento do cotidiano para o conhecimento escolar, atrelando um ao outro. A resolução de tais problemas se trata de apresentar situações que exigem dos alunos uma atitude para buscar suas próprias respostas (POZO, 1998). Muito similar às oficinas desenvolvidas, o intuito é o de ensinar os alunos a buscar a resposta para os problemas propostos, ao invés de aguardar uma resposta já pronta, descrita por um livro utilizado como base ou pelo próprio professor. Nesse contexto, o trabalho de resolver problemas não é somente do aluno, mas sim uma colaboração entre ele, seus colegas e o professor. Papel deste último o de fornecer estratégias para a resolução dos problemas, formular pesquisas bibliográficas, modificando os

problemas propostos para atender a realidade do aluno e apresentar tais problemas com uma linguagem simples e clara.

Resultados

É comum cometer erros durante a escrita de códigos, dado o alto grau de detalhismo que essa atividade exige. Durante as oficinas, tais erros foram observados por diversas vezes, mas isso não se tornou barreira que afetasse a motivação dos alunos em resolver as tarefas propostas. Acreditamos que essa dificuldade inicial seja tão mais facilmente ultrapassada quanto mais simples for a sintaxe da linguagem escolhida, tornando Python uma linguagem apropriada para esse contato inicial com programação.

Durante as aulas também foi observado que escrever o código com sintaxe correta não implica ter entendido corretamente a física envolvida no problema proposto. Para corrigir os alunos, eles são incentivados a pensar criticamente sobre o resultado obtido e sobre o processo que utilizaram para a sua obtenção. Observa-se que esse é processo importante, pois a programação, por si só, não é o mesmo que pensamento computacional (Wing, 2006). É necessário que o aluno desenvolva senso crítico sobre o que está produzindo, sendo o professor agente estimulador da significação do conhecimento. Atividade particularmente útil nesse processo é solicitar aos alunos que descrevam o modelo construído e dialogar com os pares e professores para verificar a consistência do modelo.

Em computação há diversas formas de um problema ser resolvido, utilizando-se algoritmos diferentes que forneçam respostas semelhantes. Tal liberdade é incomum na resolução da maior parte dos problemas estudados nas sala de aula, o que é um dos aspectos mais interessantes em se utilizar a computação como ferramenta pedagógica. Criar algoritmos para resolver problemas torna-se um desafio e o aluno tem a possibilidade de usar a sua criatividade para isso.

Como previsto, os alunos desenvolveram uma maior compreensão dos conteúdos de física e computação, mas também observou-se, na maior parte dos alunos, uma melhora na qualidade da sistematização e divisão em etapas dos problemas. Tal conhecimento será útil durante a vida acadêmica dos alunos, quando o processo de obtenção de resultados exige executar tarefas árduas e tediosas, e que sem ferramentas computacionais, eles têm declínio no interesse pela tarefa e disciplina (AMORIM, 2009).

Os métodos tradicionais de ensino têm papel fundamental para o êxito dessa atividade, tendo o papel de complementá-la. Escrever algoritmos para resolver cálculos oferece uma outra abordagem metodológica de ensino, e nova oportunidade para reelaborar o conteúdo aprendido pelos métodos tradicionais. E, porque escrever um algoritmo implica em reestruturar um saber, surge espaço para que surjam novas dúvidas enquanto se cria novas estruturas mentais para explicar e procedimentar o conhecimento. Dominar essa técnica também permite abrir portas para agilizar a obtenção de resultados, tornando-as uma poderosa ferramenta a ser empregada no ensino.

Firpo (2012), baseado em estudos de vários países e de seus respectivos métodos educacionais, afirma que a utilização de computadores podem ter efeito negativo no ensino. Em algumas das oficinas esteve presente um aluno com pouco interesse nas aulas, utilizando-se do acesso

irrestrito à internet para acessar mídias sociais e vídeo, servindo-se do computador como instrumento de lazer, ao invés de ferramenta de estudo.

Abaixo, o depoimento de dois alunos sobre a sua evolução durante o curso:

Aluno 1: “As aulas de Python foram bem produtivas. Há um tempo eu vinha tentando aprender programação e as minhas tentativas foram um pouco frustrantes. Depois que começamos a ter aulas com o Artur, o meu conceito sobre "programar" mudou totalmente. Não vou dizer que hoje em dia já me considere um programador fluente, porque não tivemos encontros suficientes para que isso pudesse acontecer. Mas posso dizer que aprendi muita coisa com o Artur, que sempre fazia com que Python parecesse a coisa mais fácil do mundo de se aprender (o que é quase isso). Acredito que o mais importante que eu absorvi dessas aulas não foi exatamente como se programar com Python, mas a sua utilidade em si. O Artur transpareceu bastante nos contando como essa linguagem de programação era útil em sua vida dando exemplos práticos, como trabalhos de faculdade que se resolviam mais rapidamente a partir de um simples software feito em Python, o que me deixava ainda mais animado a aprender a programar. Agora que vou ingressar na vida universitária tenho certeza que esse conhecimento fará diferença, e que Python ainda vai me evitar algumas dores de cabeça”.

Aluno 2: “Eu não sabia nada de Python, realmente zero. Mal tinha ouvido falar de Python. Então eu diria que a sua aula funcionou do jeito que deveria ser feito, uma boa aula. Divertida até em alguns momentos, que ajuda na fluidez da aula. Realmente, você parecia que tinha a aula toda programada, já tinha feito protótipos em casa e já tinha tudo certinho lá pra usar. Acho que até pelo fato de sermos menos pessoas abria mais espaço pra nossas dúvidas e conseqüentemente você ser mais específico e atencioso. Mas eu diria que foi bom, realmente tudo que eu sei de Python hoje foi você que ensinou. Mesmo que não tenhamos aprendido a fazer gráficos, ajudou a despertar um pouco de interesse também”.

Conclusão

Propusemos uma metodologia para inserir a computação dentro da rotina de ensino de Física no ensino médio, sendo Python uma linguagem que mostrou-se eficaz para estimular o desenvolvimento do pensamento computacional e a aprendizagem da ementa de física, oferecendo ferramentas ao professor para a significação do conteúdo. Todos os que participaram das atividades apresentaram avanços em seus conhecimentos de computação e compreenderam como ela pode estar associada às disciplinas de Física e Matemática, sendo que alguns alunos mostraram interesse em dar continuidade ao estudo de programação em Python. É necessário ainda que projetos semelhantes se multipliquem em outras áreas do saber e outras metodologias sejam desenvolvidas, além do treinamento de professores para a sua utilização dessas tecnologias, onde o aluno tem a liberdade de construir modelos e criar e testar seus próprios algoritmos.

Agradecimentos

Os autores agradecem a instituição Colégio Pedro II, ao Observatório de Educação (OBEDUC), ao Laboratório Interdisciplinar Educação em Ciências (LIEC), aos alunos que participaram das

oficinas, ao professor Alan Freitas Machado, ao Instituto de Física Armando Dias Tavares e a Universidade do Estado do Rio de Janeiro.

Referências Bibliográficas

AIKEN, J. M., et al. **Understanding student computational thinking with computational modeling.** *arXiv preprint arXiv:1207.1764* (2012).

AMORIM, E. S.. **Pedagogia no Ensino Superior: uma reflexão inicial.** Disponível em: <<http://revista.fundacaoaprender.org.br/index.php?id=130>> Acesso em: 29 de março de 2015

CAPPELLETTI, I. F. Avaliação do Programa “Um Computador por Aluno” (PROUCA): uma proposta inovadora em políticas públicas. Programa de PósGraduação em Educação da Pontifícia Universidade Católica de São Paulo/PUC – SP. **Revista e-curriculum**, São Paulo, V. 8, n. 1, 2012.

CARRANO, P.. MEC: **Programas de inclusão estão entre os maiores do mundo.** Disponível em: <<http://www.emdialogo.uff.br/materia/mec-programas-de-inclus%C3%A3o-digital-est%C3%A3o-entre-os-maiores-do-mundo>>. Acesso em: 28 de abril de 2015

FIRPO, S.; DE PIERI, R. Avaliando os efeitos da introdução de computadores em escolas públicas brasileiras. **Revista Brasileira de Inovação**, V. 11, 2012, p. 153-190.

OECD, PISA 2012 **Results in Focus**, 2013

OECD, PISA 2009 **Results: Executive Summary**, 2010

PAIS, L. C. **Didática da matemática: uma análise da influência francesa.** 2. Ed. Belo Horizonte: Autêntica, 2008.

PEREIRA JÚNIOR, José Carlos Rocha ; RAPKIEWICZ, Clevis Elena. O Processo de Ensino e Aprendizagem de Algoritmos e Programação: Uma Visão Crítica da Literatura. In: **III Workshop de Educação em Computação e Informática do Estado de Minas Gerais**, Belo Horizonte-MG, 2004.

POZO, J.(org). **A Solução de Problemas: Aprender a resolver, resolver para aprender.** Porto Alegre: Artmed, 1998.

PROINFO; **Adesão ao Proinfo utilizando o sistema SIGETEC**, Ministério da Educação. Disponível em: <<http://portaldoprofessor.mec.gov.br/storage/materiais/0000013473.pdf>> Acesso em 28 de abril de 2015.

Wen, X., et al. "Exploration on Cultivating Students' Abilities Based on Python Teaching Practice." **2nd International Conference on Teaching and Computational Science.** Atlantis Press, 2014.

Wing, J. M. "Computational thinking and thinking about computing." **Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences**, 2008, p. 3717-3725.

Wing, J. M. "Computational thinking". **Communications of the ACM**, 2006, p. 33-35.