

# UM ESTUDO DE CASO SOBRE O ENSINO DE PROGRAMAÇÃO EM PYTHON PARA ADOLESCENTES

Daniela Turossi da Rocha<sup>1</sup>

## RESUMO

O presente trabalho tem como base um estudo de caso sobre o ensino de programação na linguagem *Python* para adolescentes de uma escola pública. Assim é investigado o porquê da escolha da linguagem *Python* como primeira linguagem e explicado a didática escolhida para as aulas. Por fim são analisadas as observações realizadas durante o período das aulas, bem como os resultados do questionário que os estudantes responderam.

Palavras-chave: Programação; Python; Ensino de programação para adolescentes

## ABSTRACT

This work is based on a case study on teaching programming in the Python language to teenagers at a public school. Therefore, the reason for choosing Python as the first language is investigated and the teaching method chosen for the classes is explained. Finally, the observations made during the class period are analyzed, as well as the results of the questionnaire that the students answered.

Palavras-chave: Programming; Python; Teaching programming to teenagers

## 1 INTRODUÇÃO

A programação representa um conjunto de instruções escritas em um computador para resolver um problema, esta pode ser realizada em diversas linguagens. Aprender uma linguagem de programação é o mesmo que aprender uma nova língua, isto é, os brasileiros têm o português como língua nativa. A partir do português é possível aprender novas línguas, como inglês e espanhol. Da mesma forma quando se aprende programação é escolhida uma linguagem inicial (usualmente *Python* ou C) e nessa linguagem aprende-se a lógica de programação. Então, após ter um bom entendimento de lógica de programação é possível aprender a programar em outras linguagens, atentando para as particularidades de cada uma.

Este trabalho consiste no estudo do ensino de programação na linguagem *Python* para estudantes do ensino fundamental nos anos finais. Será investigado como introduzir o ensino de programação para adolescentes, tendo em vista as limitações das escolas públicas no que tange a infraestrutura necessária para esse tipo de aula. Ademais, também tem como objetivo refletir sobre a didática a ser aplicada para desenvolver a programação e analisar os benefícios do ensino de programação.

<sup>1</sup>Discente do Curso de Ciência da Computação da Universidade La Salle- Unilasalle, matriculada na disciplina Trabalho de Conclusão de Curso ||. E-mail: daniela.202010030@unilasalle.edu.br, sob a orientação Prof. Mozart Lemos de Siqueira. E-mail: mozart.siqueira@unilasalle.edu.br. Data de entrega: 01 Dezembro. 2023.

## 2 DESENVOLVIMENTO

De acordo com a Associação das Empresas de Tecnologia da Informação e Comunicação e de Tecnologias Digitais (Brasscom, 2021) “a demanda brasileira por profissionais nas áreas de software deverá alcançar 797 mil vagas até 2025”. O mercado de trabalho busca por agentes que cumpram diversos requisitos e contem com um bom *background* técnico para suprir o déficit de profissionais qualificados.

Para isso, o ensino de programação quando jovem é fundamental, já que provê o entendimento de lógica de programação e algoritmos, que são a base para o estudo nesse ramo. Dessa forma, os jovens sairiam do ensino médio, já com conhecimento básico em computação, podendo então aprofundar os estudos com um curso técnico ou faculdade. Assim, o mercado de trabalho teria maior facilidade de encontrar profissionais qualificados e suprir a demanda.

### 2.1 Pensamento computacional

O surgimento do conceito de pensamento computacional ocorreu com o escritor Seymour Papert , autor do livro *Mindstorms: Children, Computers, and Powerful Ideas*. O autor discorre sobre como o pensamento computacional é uma grande ferramenta para a resolução de problemas. Dessa forma, esse termo representa uma estratégia para solucionar problemas, dividindo em etapas, como a criação de algoritmos na computação.

Conforme Steve Jobs (2013): “Todos neste país deveriam aprender a programar um computador, pois isto ensina a pensar”. O pensamento computacional pode ser desenvolvido através do ensino de programação, seja por meio de uma linguagem visual como o *Scratch* ou escrita como o *Python*. Ambas são capazes de trabalhar condições, estruturas de repetições, entre outros tópicos que podem ser usados para construir aplicações e resolver problemas. Dentre os benefícios de desenvolver essa habilidade está a criatividade, já que podem existir diversas formas de resolver um desafio. Por fim o raciocínio lógico que auxilia a organizar a solução de um problema de forma estruturada.

### 2.2 Python como primeira linguagem de programação

A linguagem *Python* é considerada uma das melhores linguagens para iniciantes (Jason R. Briggs, 2013). Em 2022, em uma pesquisa realizada pela GitHub, essa linguagem foi considerada a 2º mais popular. Dentre as empresas famosas que usam o Python em sua aplicação pode-se citar a Netflix, Google e Uber. Utilizando *Python* é possível programar para o desktop, mobile e web, o que contribui para sua versatilidade. Além disso, como ponto positivo pode-se destacar que existem diversas bibliotecas (conjunto de subprogramas que garantem maior agilidade) disponíveis, bem como extensivas documentações e ampla comunidade para auxiliar.

Quanto ao ensino da linguagem *Python* existe muito material de ensino disponível gratuitamente. Por exemplo, no *Youtube* há o “Curso em vídeo” do professor Gustavo Guanabara que ensina a programar em *Python* para iniciantes por meio de aulas descontraídas e com várias atividades propostas aos estudantes. O

estudo de caso desse artigo baseou a didática das aulas conforme as apresentadas pelo professor Guanabara.

Segundo SOUZA e col. (2016) as principais dificuldades dos estudantes quanto a aprender programação são: “Aprender os conceitos de programação, a aplicação desses conceitos durante a construção de programas e a falta de motivação entre os alunos na realização da atividade de programação”. Dessa forma, o uso do Python foi escolhido para facilitar o entendimento dos conceitos relacionados a programação. Com essa linguagem é possível escrever códigos com menos linhas, o que garante uma curva de aprendizado pequena e ajuda a aplicar o conhecimento.

### **2.3 Lógica de programação e algoritmos com adolescentes**

Pode-se definir um algoritmo como um conjunto de instruções que tem como finalidade executar uma determinada tarefa, por exemplo, poderia determinar que a tarefa é informar se o número é par ou ímpar. Assim, mentalmente se sabe que o número que é divisível por 2 é par, sendo possível construir um código na linguagem desejada para realizar essa verificação. Algoritmos também podem ser encontrados no cotidiano. Um exemplo clássico é a receita culinária, ou seja, para realizar a receita é necessário seguir uma sequência de passos finita.

Já a lógica de programação é a forma como um algoritmo é estruturado, em sua devida linguagem. Não existe uma idade exata para começar a programar. É indicado ter noções básicas de matemática apenas. Um ponto importante a se considerar é a motivação dos alunos para começar no aprendizado de programação. Nesse cenário, cabe destacar o poder que a gamificação tem de aumentar o interesse.

Recentemente, Fabrícia Py Tortelli Noronha (2018) discorreu sobre a aprendizagem associada à gamificação e durante a investigação constatou como a gamificação no ambiente escolar colaborou para o engajamento dos alunos nas atividades. Assim, a criação de desafios e jogos durante o ensino de programação é importante para acrescer o interesse dos alunos.

Criada nos Estados Unidos a *Code.org* é uma organização sem fins lucrativos cujo objetivo é que qualquer estudante tenha a oportunidade de aprender ciência da computação nas escolas. Esta provê educação a mais de 80 milhões de estudantes. Além disso, existe uma parceria com escolas para adicionar a Ciência da Computação como disciplina curricular. Quanto à didática os alunos são incentivados a programar animações e outros desafios lúdicos. Essa organização é um exemplo real de que é possível inserir o ensino de programação no dia a dia dos estudantes e desde cedo construir esse conhecimento ao mesmo tempo que incentiva os alunos a seguir uma profissão na área da tecnologia.

Atualmente, no Brasil são poucas escolas de Educação Básica que ensinam programação. Em sua maioria são as escolas particulares que adicionam robótica e outras matérias com relação à computação, já que tendem a ter maior orçamento e infraestrutura. A respeito da introdução da programação na escola:

Respeitando-se os estágios de desenvolvimento intelectual das crianças, é possível incluir o ensino de programação de computadores nas escolas sem causar prejuízo nenhum ao aluno e trazer vários benefícios ao processo de ensino e aprendizagem de outras disciplinas GERALDES ( 2014, p.12)

Então, é possível inferir que o aprendizado de programação é capaz de trazer benefícios aos estudantes na sua vida cotidiana e ajudar em outras disciplinas, ao mesmo tempo. O desenvolvimento da criatividade, trabalho em equipe e lógica são apenas algumas das habilidades que podem ter crescimento.

## 2.4 Estudo de caso

Para o desenvolvimento deste trabalho foram ministradas aulas de programação em Python em uma escola pública do município de Canoas. A turma foi composta de 24 (vinte e quatro) alunos do 8º ano da escola, com idade aproximada de 13 (treze) anos. Esse projeto foi aprovado pelo Comitê de Ética em Pesquisa, localizado na Av. Victor Barreto, 2288 – Canoas, RS. CEP: 92010-000 - prédio 5, 2º andar. Para a aprovação foram reunidos diversos documentos e submetidos na Plataforma Brasil para avaliação.

A escola tem em torno de 1100 alunos de 1º a 9º ano e está inserida em uma comunidade carente. Quanto à infraestrutura conta com apenas uma sala de informática em que os alunos podem usar os *chromebooks*. A sala de informática foi reservada e usada durante todo desenvolvimento do projeto. As aulas ocorreram uma vez por semana em um período aproximado de 2 meses, entre 16 de agosto de 2023 até o dia 27 de setembro de 2023. Cada aula ocorreu no período de matemática e teve duração em média de 1 hora.

### 2.4.1 Didática e ambiente usado para programação

Foi preparado um material didático no formato de *powerpoint* com os principais tópicos para o aprendizado de lógica de programação na linguagem *Python*. A primeira parte da aula é no formato expositivo explicando o conteúdo e usando o quadro branco para escrever os principais pontos. Na segunda parte da aula os alunos são desafiados a tentar resolver exercícios práticos. Então na aula seguinte ocorre a correção das atividades e checagem de dúvidas e só então é prosseguido com o próximo conteúdo.

Os seguintes conteúdos foram trabalhados em aula:

- O que é programação?
- Características da linguagem de programação Python.
- Variáveis e tipos básicos.
- Funções específicas como `print()`, `input()` entre outras.
- Operadores aritméticos.
- Estrutura condicional.
- Estrutura de repetição.
- Listas.
- Como criar funções?

Quanto aos exercícios para praticar o conteúdo aprendido, foram focados em problemas matemáticos da vida diária como por exemplo calcular o preço com

desconto, média, área de um terreno, tabuada, entre outros problemas. Os exercícios tiveram esse foco já que as aulas ocorreram em períodos de matemática e para aproveitar conteúdos que os alunos trabalham nas aulas normais dessa matéria. Os exercícios foram originados do site da *python.org* e modificados de acordo com as necessidades. Além disso, as aulas e atividades práticas tiveram o objetivo de ajudar os estudantes a desenvolverem o raciocínio lógico, que é fundamental para organizar a solução de um problema de forma estruturada.

Os alunos tiveram a oportunidade de criar os programas e executá-los em um ambiente de desenvolvimento *online* chamado “*Online GDB*” ([https://www.onlinegdb.com/online\\_python\\_compiler](https://www.onlinegdb.com/online_python_compiler)). Esse ambiente foi escolhido com base na infraestrutura da escola, já que não foi possível baixar nenhum programa específico nos computadores. Em vista desse cenário, foi definido um site online para executar os programas que todos têm acesso de forma gratuita. Um problema notado em algumas aulas foi o próprio site não conseguir aguentar o acúmulo de acessos simultâneos e falhando. Felizmente, esse problema ocorreu poucas vezes e o site voltou a funcionar rapidamente.

Tendo em vista essa experiência no caso de a programação ser uma disciplina usual nas escolas é fundamental que haja uma infraestrutura adequada e autorização para instalar os programas necessários. Ademais com a oportunidade de baixar outros programas é possível mostrar a programação de forma mais interativa e abstrata com jogos.

#### 2.4.2 Observações das aulas

No decorrer das aulas foram realizadas observações do comportamento dos alunos e do envolvimento nas atividades propostas. Nas aulas iniciais sobre o conceito de programação os alunos estavam bem participativos. Ao aprender o primeiro comando de programação ‘*print*’ a maioria da turma teve o primeiro contato com uma linguagem de programação e ficaram surpresos com a possibilidade de interagir e comandar o computador usando *Python*. Na turma apenas dois alunos já haviam estudado alguma linguagem de programação anteriormente.

No momento dos exercícios propostos para que os alunos tentassem sozinhos foi possível tirar dúvidas de forma individual e ajudar os estudantes a corrigirem os erros e fazer o programa funcionar. Quanto aos erros, a maioria foram erros simples como errar uma letra ou faltar um caractere específico. Nesse período notou-se que alguns alunos não tinham muita familiaridade com o manuseio de um computador, como por exemplo quais teclas usar para copiar um texto ou dar *enter*. Além disso, o número grande de alunos foi um desafio para atender a todos de forma individual. Dessa forma, por motivo de tempo era possível atender alguns alunos e então na próxima aula corrigir as atividades de forma geral para toda turma.

Ademais, em uma aula específica foi enfrentado o desafio da falta de internet. Nesse dia a aula foi dada de forma expositiva mostrando o conteúdo no *power point*, escrevendo no quadro e rodando os códigos localmente. Em decorrência da falta de internet os alunos não tiveram a chance de programar do lado deles, por isso na aula seguinte foi planejado mais tempo para revisão e exercícios.

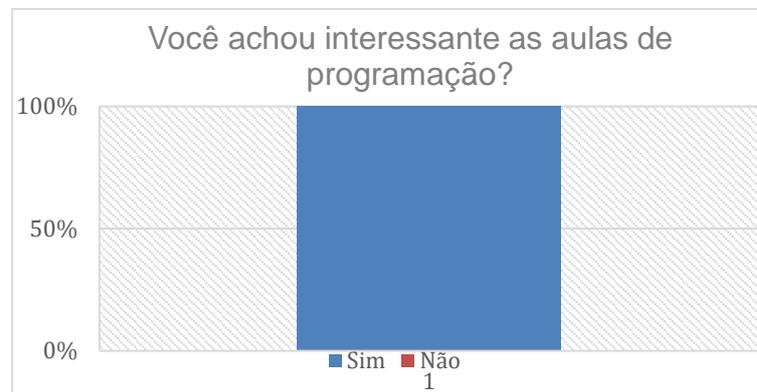
Cabe destacar que a realização de perguntas durante a aula foi produtiva e instigou os alunos a pensarem e responderem qual o resultado de tal linha de código por exemplo. Quando foi ensinado o conteúdo de estrutura de repetição percebeu-se maior dificuldade por parte dos alunos de realizar as atividades sozinhos, então os exercícios foram feitos passo a passo com a turma inteira. Na última aula foi passado o conteúdo de funções e realizado um rápido questionário online para coletar a opinião dos alunos sobre as aulas.

Ademais o professor de matemática que acompanhou as aulas compartilhou o seguinte parecer sobre o projeto: “As aulas de programação na linguagem *Python* apresentadas pela Daniela foram muito interessantes, tanto pelo conteúdo quanto pela didática/atenção que a ministrante dedicou às oficinas. Muitos estudantes se mostraram interessados em aprender sobre essa e outras linguagens de programação. Hoje em dia cada vez mais se fala sobre a necessidade de aproximar os estudantes à informática e à programação, pois as profissões do futuro terão cada vez mais estes conhecimentos como pré-requisito. Por isso acho de suma importância a inserção destas habilidades no currículo escolar. A meu ver as aulas foram muito bem aproveitadas pelos estudantes, tanto por apresentar um universo novo (o qual a maioria não tinha nenhum contato anterior) quanto para estimular o desenvolvimento do raciocínio lógico.”

#### 2.4.3 Resultados do questionário

Foi realizado um questionário opcional e anônimo na plataforma Microsoft Forms e compartilhado com os alunos. Este foi constituído por 7 perguntas no total, 5 perguntas objetivas e 2 abertas de caráter descritivas. Dos 24 participantes, 9 responderam ao formulário e contribuíram para a coleta dos dados abaixo.

Gráfico 1 - Você achou interessante as aulas de programação?



Fonte: Elaborado pela autora (2023).

Gráfico 2 - Você sentiu maior facilidade nas aulas de matemática após as aulas de programação?



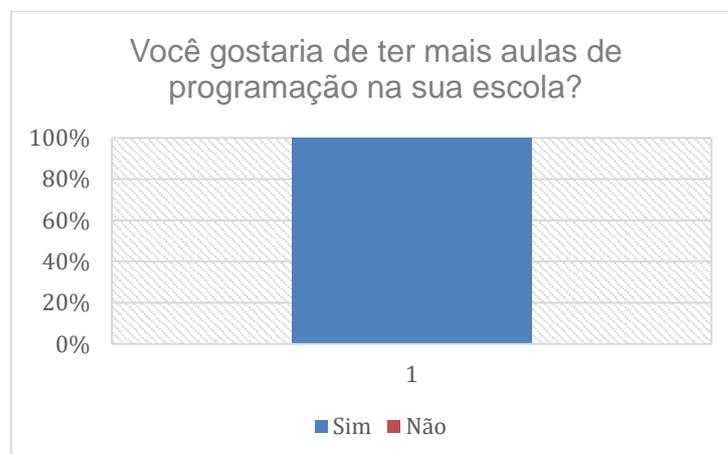
Fonte: Elaborado pela autora (2023).

Gráfico 3 - Você sentiu maior facilidade nas aulas de inglês após as aulas de programação?



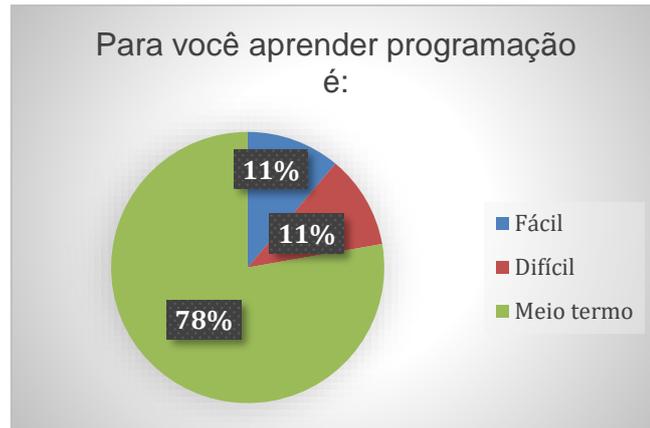
Fonte: Elaborado pela autora (2023).

Gráfico 4: Você gostaria de ter mais aulas de programação na sua escola?



Fonte: Elaborado pela autora (2023)

Gráfico 5: Para você aprender programação é:



Fonte: Elaborado pela autora (2023).

Para a pergunta “Para você quais foram as maiores dificuldades durante as aulas?” a maioria dos alunos mencionou a dificuldade de decorar os nomes de funções e como montar o código.

Na última pergunta “O que você mais gostou durante as aulas de programação?” os alunos mencionaram a aula sobre listas, como criar funções e o aprendizado de programação em geral.

### 3 CONSIDERAÇÕES FINAIS

Portanto, com base nesse estudo de caso, conclui-se que tendo em vista as dificuldades dos alunos seria útil primeiramente aulas de noções básicas de informática antes de partir para o ensino de programação. Além disso, com o objetivo de os alunos terem mais contato com o conteúdo para praticarem mais e conseqüentemente decorar as funções seria imprescindível que a disciplina tivesse no mínimo dois períodos durante a semana.

Ademais para a realização de aulas de programação é fundamental uma boa infraestrutura com computadores suficientes para todos os alunos, adicionalmente, a autorização para baixar programas adicionais é importante para diversificar o aprendizado com jogos e conteúdos mais abstratos. Por fim, o resultado do questionário comprovou que os estudantes participantes não só acharam interessante as aulas, mas gostariam de ter mais aulas de programação em sua escola. Quanto aos benefícios percebidos pelos alunos mais da metade confirmou maior facilidade em matemática e inglês após o início das aulas de programação em *Python*.

Dessa forma, ainda há um longo caminho para implementar o ensino de programação como uma disciplina regular no Brasil, como a qualificação e regularização de professores nessa área, bem como a reestruturação dos horários das matérias. Todavia com esse estudo de caso pode-se constatar que essa é sim uma possibilidade, já que há interesse por parte dos alunos e pode trazer benefícios na vida do estudante.

## REFERÊNCIAS

GERALDES, W. B. Programar é bom para as crianças? Uma visão crítica sobre o ensino de programação nas escolas. *Texto Livre*, Belo Horizonte-MG, v. 7, n. 2, p. 105–117, 2014. DOI: 10.17851/1983-3652.7.2.105-117. Disponível em: <https://periodicos.ufmg.br/index.php/textolivre/article/view/16677>. Acesso em: 11 jun. 2023.

NORONHA, F. P. T. ; BACKES, L. Projetos de aprendizagem associados à gamificação: adaptação da prática pedagógica. In: SCHLEMMER, E.; BACKES, L.; PALAGI, A. M. M.; GUEDES, A. L. (orgs.). *O habitar do ensinar e do aprender: desafios para/na/da educação OnLIFE*. 1. ed., v. 2. São Leopoldo: Casa Leiria, 2022, p. 137-150. Disponível em: <http://www.guaritadigital.com.br/casaleiria/acervo/educacao/desafios/137/index.html>. Acesso em: 11 jun. 2023.

PAPERT, Seymour. *Mindstorms: Children, Computers, and Powerful Ideas*. New York. Basic Books, Inc., Publishers. 1980.

R. BRIGGS, Jason. *Python for Kids*. San Francisco. No Starch Press. 2013.

SOUZA, Draylson Micael de e BATISTA, Marisa Helena da Silva e BARBOSA, Ellen Francine. Problemas e dificuldades no ensino e na aprendizagem de programação: um mapeamento sistemático. *Revista Brasileira de Informática na Educação*, v. 24, n. 1, p. 1-14, 2016 Tradução. Disponível em: <http://dx.doi.org/10.5753/rbie.2016.24.1.39>. Acesso em: 11 jun. 2023.