

O caminho da Tartaruga: Python e a biblioteca Turtle na educação

ARTIGO

Francisco Davi Camilo Ribeiroⁱ

Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Fortaleza, CE, Brasil

Francisca Risolene Fernandes Rochaⁱⁱ

Universidade Federal do Ceará, Fortaleza, CE, Brasil

Vitória Chérida Costa Freireⁱⁱⁱ

Secretaria Municipal de Educação de Fortaleza, Fortaleza, CE, Brasil

1

Resumo

As novas tecnologias modificaram as formas de viver no século atual, gerando impactos no ambiente educativo, o qual pode ser incrementado com novas possibilidades metodológicas, a fim de desenvolver a aprendizagem discente. Dessa maneira, o objetivo deste artigo é discutir como a aplicação da metodologia de ensino proposta pela linguagem Logo facilita o ensino da programação e de alguns fundamentos matemáticos, mesmo quando voltado para pessoas com pouco conhecimento prévio acerca dessa linguagem. O estudo, de abordagem qualitativa, foi desenvolvido mediante interações no ambiente Python e em sua biblioteca Turtle, que permitiram a execução de alguns comandos de programação para exemplificar como a referida linguagem é pertinente para o ensino de programação e de algumas noções matemáticas. Evidenciou-se que tal abordagem possibilita uma aprendizagem mais construtiva, interativa e significativa, se comparada às formas tradicionais de ensino, uma vez que o sujeito aprendiz é instigado a construir o seu próprio aprendizado com base nos seus erros, por meio de uma relação interativa dentro do ambiente, proporcionando um aprendizado com significado. Conclui-se que a linguagem de programação é um mecanismo importante para diversificar a escolarização formal, principalmente quando se trata do ensino de programação e de matemática.

Palavras-chave: Novas tecnologias. Linguagem de Programação. Ensino. Logo.

The Turtle's Path: Python and the Turtle Library in Education

Abstract

New technologies have changed the way we live in the current century, generating impacts on the educational environment, which can be enhanced with new methodological possibilities in order to foster student learning. Thus, the objective of this article is to discuss how the application of the teaching methodology proposed by the Logo language facilitates the teaching of programming and some mathematical fundamentals, even when aimed at individuals with little prior knowledge of this language. The study, which adopts a qualitative approach, was developed through interactions in the Python environment and its Turtle library, which allowed the execution of several programming commands to illustrate how

this language is relevant to the teaching of programming and some mathematical notions. It was evident that this feat enables more constructive, interactive and meaningful learning, when compared to traditional forms of teaching, since the learning subject is encouraged to construct their own knowledge based on their mistakes, through an interactive relationship within the environment, providing meaningful learning. It is concluded that the programming language is an important mechanism for diversifying formal schooling, especially when it comes to teaching programming and mathematics.

Keywords: New technologies. Programming language. Teaching. Logo.

2

1 Introdução

Cada período histórico, levando em conta as suas especificidades, demanda um tipo de formação educativa, de tal maneira que, ao longo do tempo, percebemos haver uma relação direta entre educação e sociedade (Durkheim, 1978). Assim, tendo em vista que a contemporaneidade é marcada por avanços tecnológicos nas mais variadas áreas, o processo educacional não sai ileso desse clima de transformação e é diretamente por ele afetado, seja na proposição e implementação de novas formas (metodologias) de ensinar e aprender, seja nos desdobramentos (positivos e negativos) que, mesmo inconscientemente, as novas tecnologias têm gerado nos contextos das salas de aula (Castells, 2002).

Nessa perspectiva, de acordo com Moran (2009), a educação formal do século XXI, ou seja, aquela desenvolvida em espaços voltados especificamente para esse fim, é um processo demasiadamente diferenciado de todos os modelos de educação já existentes ao longo da história, dadas as particularidades do contexto inaugurado com o surgimento das Tecnologias da Informação e Comunicação (TICs). Tendo em vista a impossibilidade de desconsiderar esse cenário, já que ele está presente na vida de todos, tanto de professores quanto dos alunos, interessa pensarmos em maneiras de fazer um bom uso das novas tecnologias, de modo a tomá-las como aliadas da aprendizagem (Castells, 1999). É partindo desse princípio que vem sendo ressaltada uma variedade de ambientes virtuais e linguagens de programação interativas que corroboram a

aprendizagem (Calvo, 2016), a exemplo da Logo, que pode ser utilizada para o ensino da programação e de alguns conceitos matemáticos para um público diverso.

3

A Logo é uma linguagem de programação desenvolvida por Seymour Papert, Cynthia Solomon e Wally Feurzeig no fim da década de 1960, no Massachusetts Institute of Technology (MIT), voltada para o ensino de crianças, adolescentes e até adultos, com o objetivo de instigar os seus usuários a aprender explorando, investigando, criando e descobrindo por si mesmos não apenas conceitos de computação, mas também matemáticos (Coluci, 2022).

Em linhas gerais, podemos compreender o Logo como um ambiente no qual se apresenta um robô que obedece aos comandos do usuário, pois ele faz uso de uma linguagem considerada interativa, ou seja, que vai se construindo por meio dos direcionamentos recebidos, em um processo de construção (Pinheiro *et al.*, 2024). Tal linguagem é propícia ao desenvolvimento do ensino de programação e de alguns conceitos matemáticos, tendo em vista que o usuário, ao interagir nesse espaço, vai construindo o seu conhecimento mediante a relação interativa que estabelece com o robô, que, no caso, é a Tartaruga, a qual, por sua vez, recebe o conhecimento transmitido pelo sujeito que com ela interage. Assim, percebe-se a relação direta entre o ambiente Logo e a escolarização formal, mais especificamente no que diz respeito ao ensino de programação e de conceitos matemáticos, que são dois campos do conhecimento frequentemente considerados difíceis para muitos, o que aponta para a importância de serem utilizados mecanismos que tornem a aprendizagem mais facilitada (Calvo, 2016).

A Logo possui uma linguagem própria e ambientes de programação voltados especialmente para ela. Porém, devido ao desuso da linguagem e à incompatibilidade de alguns desses ambientes, faremos uso da linguagem Python e da biblioteca Turtle, que é nativa da linguagem e foi implementada como uma ferramenta de desenho geométrico baseada em Logo com todas as suas funcionalidades adaptadas para o uso com a linguagem Python.

A ideia de abordar esse tema não é invalidar outras metodologias de ensino ou substituí-las, mas mostrar que o conhecimento pode ser construído e se desenvolver de

formas distintas daquela maneira tradicional que recorrentemente se concretiza nos ambientes educacionais formais (Saviani, 2007). Nessa perspectiva, com o recurso tecnológico ora enfatizado, há a possibilidade de desenvolver o ensino-aprendizagem de maneira mais interativa, cativando o estudante para que ele não fique preso apenas aos métodos mais tradicionais de ensino e aprenda por meio daquele que, possivelmente, seja o método mais utilizado ao estudar linguagens de programação em geral: o aprendizado com base nas suas próprias experiências, introduzindo, assim, de forma mais leve, o ensino da programação (Parpet, 1980).

Dessa forma, parte-se do seguinte questionamento: como a linguagem Logo contribui para o aprendizado de programação e de algumas noções matemáticas, voltado para jovens, crianças e adultos, tornando esse processo mais prazeroso? Portanto, o objetivo é discutir como a aplicação da metodologia de ensino proposta pela linguagem Logo facilita o ensino da programação e, por conseguinte, de alguns fundamentos matemáticos, mesmo quando voltado para pessoas com pouco conhecimento prévio acerca dessa linguagem. Para tanto, utiliza-se a linguagem Python e a sua biblioteca Turtle para produzir exemplos em formato de códigos de programação, demonstrando a ilustração de figuras geométricas que podem ser utilizadas no ensino da matemática, como resultados da execução de códigos e do trajeto da Tartaruga.

A relevância desse feito consiste em apresentar uma possibilidade diferenciada para facilitar o ensino de programação e de algumas noções matemáticas, seja para crianças, seja para jovens ou adultos. De tal maneira, dialoga-se com os campos da programação e o educacional, sendo este último o que possui maior realce no presente estudo, já que é sob o intuito de melhor qualificar a relação de ensino-aprendizagem que foi pensado o ambiente Logo na sua interface com a educação. Portanto, ressalta-se o potencial das novas tecnologias digitais da informação e da comunicação para o desenvolvimento da aprendizagem no ambiente de escolarização formal (Moran, 2009).

Visando à organização do texto para um melhor entendimento por parte do leitor, o manuscrito encontra-se estruturado da seguinte maneira: primeiro, temos a presente introdução, que apresenta a temática abordada, esboçando do que se trata o ambiente

Logo, bem como explicitando o objetivo, o questionamento e a relevância do estudo. Na sequência, temos a discussão sobre o percurso metodológico empregado para atingir o objetivo proposto e, adiante, a apresentação da Logo como linguagem que potencializa a aprendizagem da programação e da matemática, discutindo a sua pertinência para o ambiente de ensino por meio da apresentação do caminho da Tartaruga. Por fim, as considerações encerram o texto, retomando o objetivo e apresentando as principais reflexões oriundas das discussões realizadas.

2 Metodologia

O estudo possui abordagem qualitativa (Minayo, 1994), tendo em vista que se efetiva por meio do olhar interpretativo, sob o viés da subjetividade ligada ao universo das operacionalizações variáveis, não interessando a quantificação de dados ou a busca por uma verdade absoluta e inquestionável. Segundo explica Minayo (1994), a abordagem qualitativa lança luz ao entendimento de fenômenos sociais a partir da análise minuciosa de objetos de estudo que não seriam comprehensíveis pela ótica positivista, podendo tal objeto ser investigado a partir de variadas perspectivas. Por isso, ressalta-se a influência dessa abordagem de pesquisa no presente manuscrito, cuja centralidade é a discussão de questões educacionais na interface com as novas tecnologias, especificamente no que diz respeito à linguagem Logo para o ensino de programação e matemática.

Para atingir esse intento, como a linguagem Logo é bastante antiga e já se encontra, praticamente, em desuso, fez-se uso do ambiente de desenvolvimento integrado (IDE) da linguagem de programação Phyton e da biblioteca Turtle, por meio da qual foram simulados exemplos de como o Logo contribui para a aprendizagem da programação e da matemática. Especificamente, utiliza-se a biblioteca Turtle, que se trata de uma ferramenta na qual é possível fazer desenhos e animações simples ao mover uma tartaruga virtual.

Dentre tantas outras possibilidades, foram realizadas ilustrações de figuras geométricas planas (reta, triângulo, retângulo e círculo) por meio de códigos de programação. Os exemplos dos códigos e das figuras a eles relacionados foram

documentados no formato de imagens e seguem no corpo do trabalho, a fim de melhor qualificar as discussões e alcançar o objetivo ora traçado.

6

Por fim, no que diz respeito aos aspectos éticos, interessa esclarecer que o estudo se adequa ao que determina a Resolução nº 510, de 7 de abril de 2016, mecanismo elaborado pelo Conselho Nacional de Saúde (CNS) para instituir as normas a serem seguidas em pesquisas nas áreas das Ciências Humanas e Sociais. Isso porque, aqui, não se adotam seres humanos como participantes do estudo, o que dispensa que a pesquisa seja submetida ao comitê de ética.

3 A filosofia Logo e as suas potencialidades para a educação

A linguagem de programação Logo é composta de uma Tartaruga gráfica, a qual se constitui em um robô pronto para interagir com os comandos escritos do usuário. Uma vez que a linguagem é interpretada e interativa, o resultado é mostrado imediatamente após o comando ser executado (Papert, 1980). Nela, o usuário aprende com os próprios erros, vivenciando e repassando esse conhecimento para a Tartaruga. Se algo está errado em seu raciocínio, tal erro é claramente percebido e demonstrado em tela pelo ambiente Logo, fazendo com que o sujeito aprendiz reflita sobre o que poderia estar errado e tente, a partir dos erros por ele experienciados, encontrar as soluções corretas para os problemas.

Portanto, trata-se de um ambiente que coloca o estudante como ser ativo no seu processo de aprendizagem – não no sentido de dispensar ou desprezar a figura do professor, mas no sentido de que é o próprio usuário quem irá buscar ativamente o conhecimento, na sua interação com a Tartaruga, que, por sua vez, irá instigá-lo a pensar. Sob esse ponto de vista, o aluno não recebe o conhecimento pronto e acabado, como acontece na perspectiva da educação tradicional que o coloca como depósito de informações que o professor, a sociedade ou o currículo julgam ser importantes (Saviani, 2007); ao contrário, ele aprende com a própria experiência ao interagir com a Tartaruga, sendo um potencial construtor da própria aprendizagem.

Assim, o Logo propõe uma metodologia de ensino que busca, por meio de uma linguagem simples, prática e aproximada da linguagem natural, facilitar a comunicação entre o usuário e o computador, proporcionando a criação de modelos gráficos por meio de formas geométricas e de raciocínio lógico. Mas o principal por trás da ferramenta Logo é que existe um propósito ainda maior, qual seja: a ideia de que o estudante deve ser um ativo construtor dos próprios conhecimentos, desenvolvendo, assim, a sua capacidade de raciocínio, uma vez que ele é levado, a partir da interação com a linguagem Logo, a refletir sobre novas formas de resolução dos problemas apresentados, aprendendo a aplicar novas abordagens a todo tempo.

Além do mais, em razão de seu fundamento alicerçado na aprendizagem construída e, portanto, com significado para o aprendiz (Ausubel, 1982), a filosofia Logo considera o erro como um importante fator de aprendizagem, que oferece oportunidades para que o estudante entenda o motivo de seu erro e busque uma nova solução para o problema, investigando, explorando e descobrindo por si próprio; ou seja, o objetivo final é uma aprendizagem que se processa por meio da descoberta. Dessa maneira:

Quando acontece um erro, este torna-se um objeto de análise para que seja identificado e reformulado, desencadeando aprendizagem e desenvolvimento. Esse processo estabelece um ciclo de descrição-depuração - reflexão-depuração, que foi implantado na programação de computadores. (Papert, 1980, p. 93).

Assim, torna-se evidente o quanto a linguagem Logo contribui para uma aprendizagem significativa, já que, conforme o trecho anterior, ao identificar um erro, ela o apresenta ao estudante para que ele o tome como objeto de análise, dando início aos processos de descrição-depuração e de reflexão-depuração, ou seja, de busca por possíveis maneiras de resolver o problema ora apresentado.

Nesse momento, podemos entender a importância da Tartaruga para a abordagem proposta pela filosofia Logo – a Tartaruga faz menção à imagem que aparece para o usuário ao interagir com a linguagem, que se assemelha a uma tartaruga, de tal maneira que ficou conhecida por esse nome dentro do universo da programação. A Tartaruga, então, apoia o usuário, deixando claro seus erros e acertos, proporcionando,

inclusive, que ele controle a velocidade com que executa os seus movimentos e permitindo, ao acompanhar a execução das ações, observar em que momento a sua lógica apresentou alguma inconsistência.

8

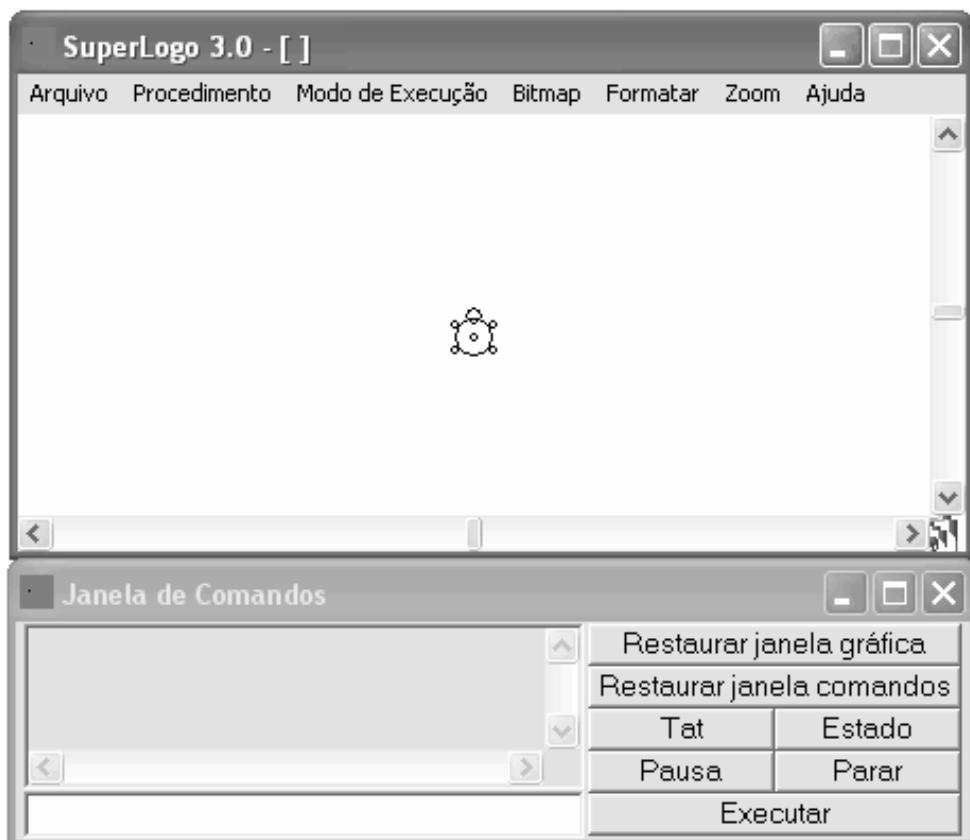
Por adquirir tais características, a linguagem Logo implementa, em alguns aspectos, a filosofia construcionista segundo a visão do próprio Papert, o que ficou conhecido como Construcionismo Informático, que estuda o desenvolvimento e o uso da tecnologia, em especial, do computador, na criação de ambientes educacionais (Papert, 1980). Tal fato demonstra a grande relevância do Logo enquanto ferramenta de ensino-aprendizagem, realçando que ela tem muito a contribuir para aprimorar o desenvolvimento educativo, em especial no âmbito da programação e da matemática, seja no ambiente da educação formal, por iniciativa dos professores, seja por interesse do próprio indivíduo que, sozinho, pode buscar aprender por meio dessa linguagem interativa (Coluci, 2022).

A filosofia Logo, então, traz em seu cerne a ideia de que o conhecimento é construído por meio das experiências do indivíduo e do ambiente no qual este está inserido, e que pode ser estimulado a melhorar e aprender mesmo quando erra. Ao aproximar esses pressupostos do campo das teorias educacionais, vemos que a forma como o conhecimento é abordado a partir da linguagem Logo tem uma forte relação com a perspectiva de desenvolvimento do conhecimento proposta por Piaget, por meio da sua teoria da aprendizagem denominada Construtivismo. Isso porque, para esse psicólogo educacional, o conhecimento é construído pelo próprio sujeito em meio às suas experiências e interações com o meio, e não depositado por outros, como ocorre na vertente de ensino tradicional (García, 2002), sendo justamente essa particularidade o que norteia o sentido da educação a partir do uso do Logo.

A linguagem de programação Logo varia de acordo com a implementação dos desenvolvedores daquela IDE em particular. Ao longo do tempo, a linguagem Logo recebeu muitas implementações de entusiastas e estudiosos da programação, inclusive no Brasil. Aqui, o ambiente Logo é o SuperLogo 3.0, que é constituído por uma tela onde está presente uma Tartaruga, o robô gráfico que vai obedecer aos comandos do usuário conforme eles são inseridos em uma linha de comando, de forma sequencial, uma janela

de comandos que recebe funções nativas da linguagem Logo e alguns botões utilizados para o envio de comandos primários, conforme imagem abaixo:

Imagen 1 - Ambiente SuperLogo 3.0



Fonte: Autoria própria (2025).

Conforme a imagem, vemos que a tela inicial do ambiente SuperLogo 3.0 se trata de um campo em branco com a imagem de uma Tartaruga ao centro e, na parte superior e inferior à direita, temos alguns comandos que podem ser utilizados pelo usuário, como “Modo de execução”, “Zoom” e “Restaurar janela gráfica”. Portanto, é um ambiente de fácil interação – tanto que é acessível a variados públicos, desde crianças até adultos.

Apesar de, inicialmente, o ambiente SuperLogo 3.0 trabalhar apenas com um comando de cada vez, é possível escrever funções personalizadas, com maior complexidade, e carregá-las com uma função específica, assim como fazemos em outras

linguagens de programação, tudo depende da opção do usuário. Assim, a linguagem Logo pode executar comandos mais complexos e o usuário pode pensar e instruir formas mais elaboradas para serem desenhadas pela Tartaruga, assim como pode fazer uso de funções de repetição e recursivas, que se apresentam como menos complexas. Desse modo, podemos entender que os variados níveis dos comandos dependem das decisões tomadas pelo seu usuário ao interagir com o ambiente e buscar resolver os problemas que podem surgir.

Dentre as funções presentes na “Janela de Comandos”, que, na imagem anteriormente apresentada, encontra-se na parte inferior do ambiente SuperLogo 3.0, temos uma função que executa os comandos digitados no prompt de comando, um botão que limpa todo o histórico de comandos passados anteriormente no prompt, e uma função de reset, que elimina todo o progresso da Tartaruga, retornando-a ao ponto de partida e limpando a tela de quaisquer desenhos realizados anteriormente. Sobre a linguagem Logo, podemos destacar, ainda, que se trata de uma linguagem de programação interpretada, ou seja, oriunda de uma ação executada por um interpretador, que verifica cada código por linha e o analisa, concretizando as instruções em tempo real ao comando, fator crucial para que haja interação com o usuário (Papert, 1980). Além disso, a linguagem mencionada possui tipagem dinâmica, forte e implícita, sendo classificada dentro dos seguintes paradigmas de programação: funcional, procedural e reflexivo.

Tendo em vista que a linguagem Logo é bastante antiga (foi criada na década de 1960), fez-se necessário executar os comandos aqui utilizados como exemplos por meio do Python e da biblioteca Turtle. Python é uma linguagem de programação desenvolvida por Guido van Rossum no fim da década de 1980, tendo a sua primeira versão lançada em 1991. Em sua ideia inicial, a linguagem Python prioriza a legibilidade do código em relação ao esforço computacional, gerando, assim, uma linguagem mais amigável ao programador e que se assemelha mais à linguagem humana. Afinal, o objetivo de van Rossum era criar uma linguagem tão fácil de compreender quanto o próprio inglês (Pinheiro *et al.*, 2025).

Ao longo dos anos, a linguagem Python foi ganhando cada vez mais popularidade devido à sua rápida curva de aprendizado, inclusive para programadores iniciantes, bem como à sua usabilidade nos mais diferentes projetos. Hoje, Python pode ser usado para a análise de dados, machine learning, programação web, dentre outros. Portanto, trata-se de uma linguagem bastante acessível e com utilidade diversificada, podendo ser manuseada por pessoas de diferentes faixas etárias.

Devido à grande curiosidade e ao interesse que a linguagem Logo despertou, principalmente entre aqueles que a utilizam como uma introdução à programação, várias ferramentas foram criadas inspiradas no ambiente Logo. Dentre elas, uma adaptação muito completa e fiel ao proposto foi implementada por padrão na linguagem Python: a biblioteca Turtle. Esta emula quase por completo o comportamento e as funcionalidades de um ambiente Logo, inclusive utilizando uma Tartaruga gráfica como o ambiente Logo tradicional, junto com uma janela gráfica na qual é possível acompanhar os movimentos da Tartaruga e os desenhos por ela realizados, sendo possível aumentar ou diminuir a velocidade desses movimentos para que o usuário acompanhe o passo a passo.

Além disso, o ambiente Python se torna ainda mais similar a um ambiente Logo tradicional quando utilizamos o ambiente padrão de desenvolvimento do Python, conhecido como Integrated Development and Learning Environment (IDLE). Portanto, justifica-se, aqui, a escolha pelo uso dessa linguagem de programação, por meio de sua biblioteca Turtle, para exemplificar como a linguagem Logo contribui para a aprendizagem matemática e para o campo da programação.

A biblioteca Turtle tem como finalidade desenhar em uma tela vazia usando o robô gráfico que chamamos de Tartaruga, a qual se assemelha à ponta de um lápis que possibilita desenhar conforme os comandos são enviados. Baseando-se em nossos comandos estruturados logicamente, é fácil perceber o leque de oportunidades que temos aqui. Mas, concentrando-se especificamente na geometria, é possível executar diversas fórmulas em formato de códigos Python para que a Tartaruga nos apresente figuras já conhecidas da geometria plana, servindo como uma prova de que os conceitos aplicados são verdadeiros.

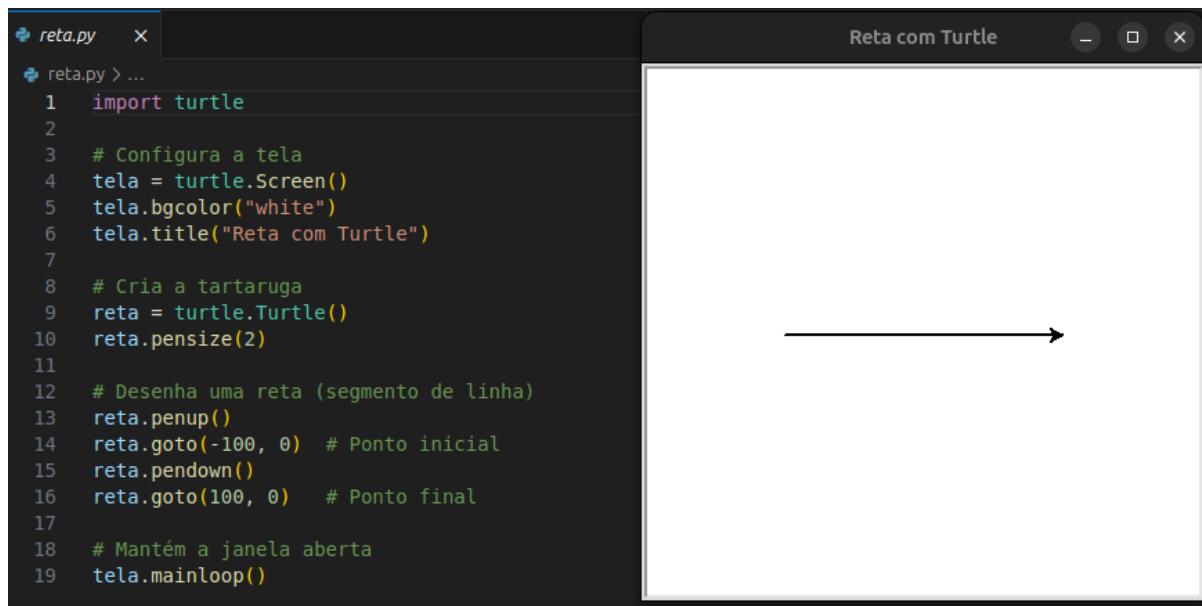
Adiante, são apresentados os códigos de programação utilizados no ambiente anteriormente descrito para que a Tartaruga desenhe algumas figuras geométricas planas, a saber: reta, triângulo, retângulo e círculo. Mas, antes disso, faz-se necessário entender do que se trata o ponto dentro dessa linguagem, já que é a partir dele que são realizados os desenhos geométricos.

12

O ponto se constitui em um ente geométrico que não pode ser definido. O que sabemos sobre ele são suas características, pois se trata da noção geométrica mais primitiva existente. Aqui, abordaremos o ponto como um pixel na tela, já que a biblioteca Turtle o considera como unidade de medida e a menor unidade de espaço que ela comprehende. Toda vez que movermos a Tartaruga pela tela, estaremos utilizando uma quantidade de pixels para informar a distância que deve ser percorrida.

Dessa forma, uma reta é a quantidade de pontos consecutivos desenhados pela Tartaruga, que vão de um ponto A até um ponto B, sem espaços não desenhados. Abaixo, temos expostos o código que instrui a Tartaruga a desenhar uma reta, bem como o desenho da própria reta:

Imagen 2 – Código que instrui a Tartaruga a desenhar uma reta



```
retap.py  x
retap.py > ...
1 import turtle
2
3 # Configura a tela
4 tela = turtle.Screen()
5 tela.bgcolor("white")
6 tela.title("Reta com Turtle")
7
8 # Cria a tartaruga
9 reta = turtle.Turtle()
10 reta.pensize(2)
11
12 # Desenha uma reta (segmento de linha)
13 reta.penup()
14 reta.goto(-100, 0) # Ponto inicial
15 reta.pendown()
16 reta.goto(100, 0) # Ponto final
17
18 # Mantém a janela aberta
19 tela.mainloop()
```

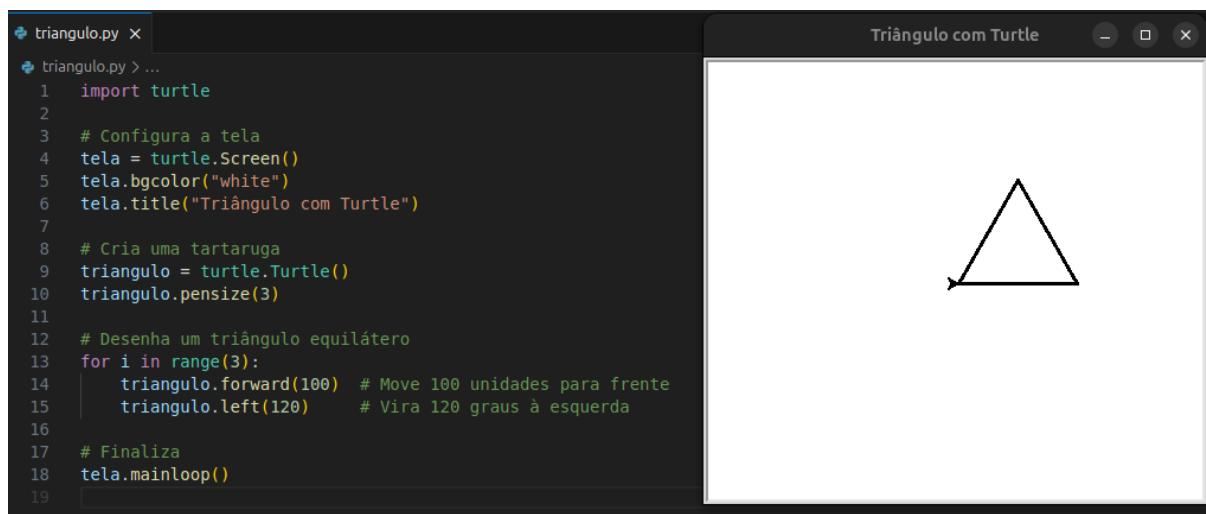
Fonte: Autoria própria (2025).

Portanto, os códigos utilizados, os quais se encontram ilustrados à esquerda (do leitor), instruem quanto à configuração da tela na biblioteca Turtle, à criação da Tartaruga e, então, ao desenho da reta, para o qual se utilizam códigos que estabelecem o ponto inicial e o ponto final.

13

Um triângulo é uma figura geométrica plana formada por três lados, três vértices e três ângulos. Para que a Tartaruga desenhe essa figura, é necessário instruí-la com os seguintes códigos:

Imagen 3 - Código que instrui a Tartaruga a desenhar um triângulo



```
triangulo.py
1  import turtle
2
3  # Configura a tela
4  tela = turtle.Screen()
5  tela.bgcolor("white")
6  tela.title("Triângulo com Turtle")
7
8  # Cria uma tartaruga
9  triangulo = turtle.Turtle()
10 triangulo.pensize(3)
11
12 # Desenha um triângulo equilátero
13 for i in range(3):
14     triangulo.forward(100) # Move 100 unidades para frente
15     triangulo.left(120)    # Vira 120 graus à esquerda
16
17 # Finaliza
18 tela.mainloop()
19
```

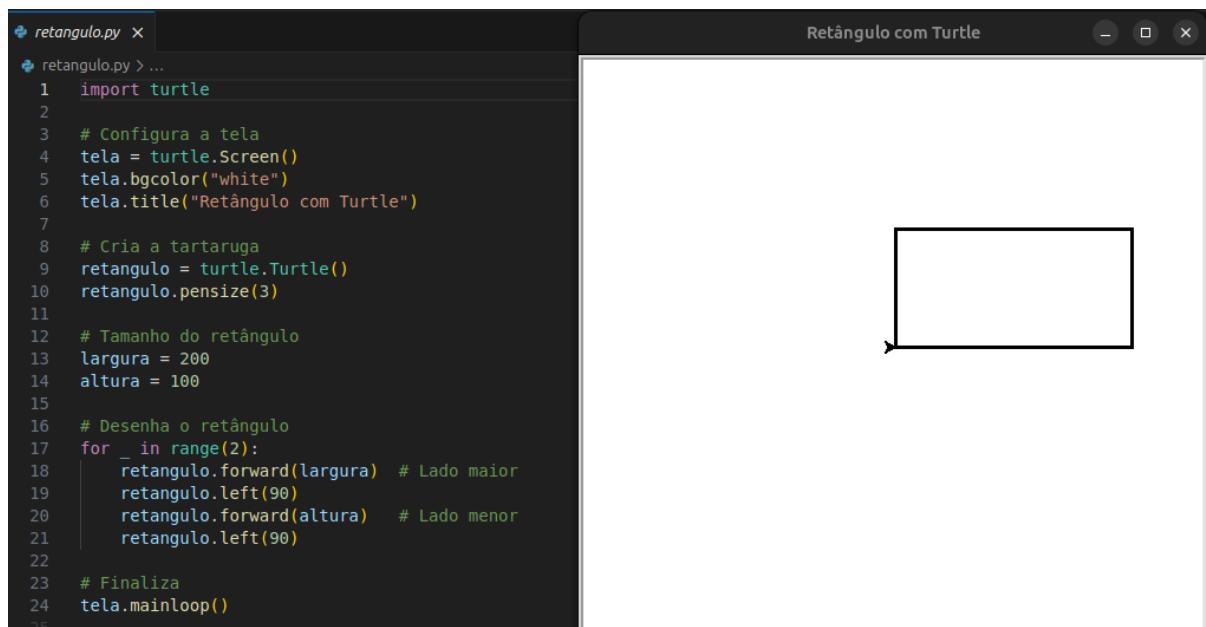
Fonte: Autoria própria (2025).

Assim, para que o triângulo seja ilustrado, é necessário que sejam dados à Tartaruga os comandos para que ela se move 100 unidades para a frente e, em seguida, vire 120 graus à esquerda. Além dessas instruções, vemos acima que a biblioteca Turtle apresenta um espaço para configurar a tela, criar a Tartaruga e finalizar.

Além da reta e do triângulo, seguindo a mesma lógica de desenhos por meio de linhas retas, podemos utilizar a Tartaruga para ilustrar um retângulo. Este se trata de uma figura geométrica plana de quatro lados (um tipo de quadrilátero), em que todos os ângulos

internos medem 90° (ângulos retos) e os lados opostos são paralelos e iguais. O código que instrui a Tartaruga a desenhar um retângulo é o seguinte:

Imagen 4 - Código que instrui a Tartaruga a desenhar um retângulo



```
14
14
1  import turtle
2
3  # Configura a tela
4  tela = turtle.Screen()
5  tela.bgcolor("white")
6  tela.title("Retângulo com Turtle")
7
8  # Cria a tartaruga
9  retangulo = turtle.Turtle()
10 retangulo.pensize(3)
11
12 # Tamanho do retângulo
13 largura = 200
14 altura = 100
15
16 # Desenha o retângulo
17 for _ in range(2):
18     retangulo.forward(largura) # Lado maior
19     retangulo.left(90)
20     retangulo.forward(altura) # Lado menor
21     retangulo.left(90)
22
23 # Finaliza
24 tela.mainloop()
25
```

Fonte: Autoria própria (2025).

À vista do exposto, fica evidente que, para desenhar o retângulo, é preciso delimitar a largura e a altura – que, no exemplo acima, são 200 e 100 – e, em seguida, efetivar o desenho da referida figura geométrica, utilizando códigos diferentes para o lado maior e para o lado menor. Além disso, a biblioteca Turtle apresenta as opções para configurar a tela, criar a Tartaruga e finalizar, como nos exemplos anteriores.

Diferentemente das ilustrações anteriores, as quais são formadas por linhas retas, a Tartaruga pode delinear, ainda, um círculo, que é uma figura geométrica plana formada por todos os pontos que estão a uma mesma distância (raio) de um ponto fixo ao meio, chamado centro. O código necessário para rabiscar o círculo é o seguinte:

Imagen 5 - Código que instrui a Tartaruga a desenhar um círculo

```
1  import turtle
2
3  # Configura a tela
4  tela = turtle.Screen()
5  tela.bgcolor("white")
6  tela.title("Círculo com Turtle")
7
8  # Cria a tartaruga
9  circulo = turtle.Turtle()
10 circulo.pensize(3)
11
12 # Desenha um círculo com raio 50
13 circulo.circle(50)
14
15 # Finaliza
16 tela.mainloop()
```

Fonte: Autoria própria (2025).

Nesse último exemplo, vemos que, além dos campos fixos (configurar tela, criar a Tartaruga e finalizar), são utilizados códigos específicos para instruir a Tartaruga a desenhar um círculo com raio de 50 graus.

À vista dos exemplos anteriormente expostos, fica evidente que a linguagem Logo – nesse caso utilizada mediante a linguagem de programação Python e de sua biblioteca Turtle – apresenta grande potencial para incrementar o ensino da programação e, ao mesmo tempo, de conceitos matemáticos. Nos casos apresentados, o usuário precisava fazer uso de códigos distintos para que fossem desenhadas figuras geométricas planas, em um movimento interativo e construtivo com o ambiente Python, o que torna a aprendizagem dinâmica e com maior significado para o sujeito aprendiz, já que ele ocupa o lugar de construtor da própria aprendizagem. Isso porque, caso o usuário utilizasse um código incorreto, teria que buscar formas de resolver o problema, sendo instigado a pensar – não recebendo o conhecimento pronto e acabado pelo ambiente de aprendizagem.

4 Considerações finais

16

O objetivo foi discutir como a aplicação da metodologia de ensino proposta pela linguagem Logo facilita o ensino da programação e de alguns fundamentos matemáticos, no presente caso, de figuras geométricas planas, mesmo quando voltado para pessoas com pouco conhecimento prévio acerca dessa linguagem. Para alcançar esse objetivo, foi realizado um estudo de abordagem qualitativa, desenvolvido por meio da linguagem Python e de sua biblioteca Turtle, considerando que a linguagem Logo é bastante antiga e já se encontra em desuso. No ambiente mencionado, foram ilustrados exemplos de códigos que reproduziram a figura de algumas formas geométricas planas, unindo a programação à matemática.

O questionamento que instigou este exercício foi “como a linguagem Logo contribui para o ensino de programação e matemática voltado para jovens, crianças e adultos, tornando esse processo mais prazeroso?”, e, ao final das exemplificações resultantes das interações no ambiente Python por meio da biblioteca Turtle, ficou evidente que as novas tecnologias, nesse caso uma linguagem de programação, são ferramentas profícias para promover uma aprendizagem significativa e interativa.

Ao executar os códigos de programação para desenhar a reta, o triângulo, o retângulo e o círculo, foi possível perceber que o usuário está imerso em um ambiente interdisciplinar, ou seja, que une o conhecimento de mais de uma área, nesse caso em particular, da programação e da matemática. Ficou explícito, dessa maneira, que o caminho traçado pela Tartaruga (fazendo uma analogia com a linguagem Logo) é um trajeto que contribui para a educação, podendo ser utilizado, inclusive, dentro da sala de aula, seja em aulas de programação, seja em aulas de matemática, para potencializar a escolarização formal.

Além disso, uma vez que a linguagem permite a aprendizagem construtiva, com base nos erros individuais, é nítido o seu potencial para o desenvolvimento da autonomia do aprendiz, que é responsável pela busca de soluções de possíveis problemas oriundos de algum erro que possa haver no código por ele utilizado. Assim, o uso da linguagem

Logo é importante para ressignificar o ambiente educacional, afastando técnicas tradicionais baseadas na figura do aluno como receptor de informações e tornando-o um ser ativo que raciocina por si próprio.

Conclui-se que as exemplificações aqui expostas e as discussões realizadas acerca da temática programação e educação ratificam a importância das novas tecnologias da informação e comunicação para o contexto educacional contemporâneo e, assim, faz-se necessário que linguagens de programação, a exemplo da Logo, sejam utilizadas como instrumentos que fortaleçam a aprendizagem de forma construtiva e significativa para o sujeito aprendiz, independentemente de sua faixa etária. Em face das potencialidades das linguagens de programação para a educação, é importante que essa discussão seja realçada em outros estudos, tomando como base outros códigos para o desenho de outras figuras.

17

Referências

- AUSUBEL, David Paul. **A aprendizagem significativa: a teoria de David Ausubel**. São Paulo: Moraes, 1982.
- BRASIL. Ministério da Saúde. Conselho Nacional de Saúde. Resolução nº 510, de 7 de abril de 2016. Dispõe sobre as normas aplicáveis a pesquisas em Ciências Humanas e Sociais. *Diário Oficial da União: seção 1*, Brasília, DF, n. 98, p. 44–46, 24 maio 2016.
- CALVO, Alfredo Hernando. **Viagem à escola do século XXI**: assim trabalham os colégios mais inovadores do mundo. São Paulo: Fundação Telefônica Vivo, 2016.
- CASTELLS, Manuel. **A era da informação**: economia, sociedade e cultura. Lisboa: Fundação Calouste Gulbenkian, 2002.
- CASTELLS, Manuel. **A sociedade em rede**. São Paulo: Paz & Terra, 1999.
- COLUCI, Vitor. Animações de conceitos da teoria de erros usando Manim/Python. **Revista Brasileira de Ensino de Física**, v. 44, p. 1–8, 2022. Disponível em: <https://www.scielo.br/j/rbef/a/YlfHKMG9B4HWKZfPtDNgPsn/?lang=pt>. Acesso em: 3 jun. 2025.
- DURKHEIM, Émile. **Educação e sociologia**. Petrópolis: Vozes, 1978.

GARCÍA, Rolando. **O conhecimento em construção: das formulações de Jean Piaget à teoria de sistemas complexos.** Porto Alegre: Artmed, 2002.

MINAYO, Maria Cecília de Souza. **Pesquisa social: teoria, método e criatividade.** Petrópolis: Vozes, 1994.

MORAN, José Manuel. **Novas tecnologias e mediação pedagógica.** Campinas: Papirus, 2009.

PAPERT, Seymour. **Logo:** computadores e educação. São Paulo: Brasiliense, 1980.

PINHEIRO, Éverton Leal; BARATA, Karla Miranda; NASCIMENTO NETO, Odemar Julião do; MARTINS, Ramon dos Santos; SUQUEIRA, Marcelo. Aplicações computacionais em física: representação espectral da função de Green com Python. **Revista Brasileira de Ensino de Física**, v. 46, p. 1–12, 2024. Disponível em: <https://www.scielo.br/j/rbef/a/hstfJYCgS6hGrybTWc5GyKy/?lang=pt>. Acesso em: 3 jun. 2025.

SAVIANI, Dermerval. **História das ideias pedagógicas no Brasil.** Campinas: Autores Associados, 2007.

ⁱ **Francisco Davi Camilo Ribeiro**, ORCID: <https://orcid.org/0009-0000-4177-3614>

Instituto Federal de Ciência e Tecnologia do Ceará, Curso de Engenharia da Computação Graduando em Engenharia da Computação pelo Instituto Federal de Ciência e Tecnologia do Ceará (IFCE), campus Fortaleza.

Contribuição de autoria: execução dos testes de programação e elaboração da primeira escrita do texto.
Lattes: <http://lattes.cnpq.br/9150156436027825>
E-mail: davicamiloribeiro@gmail.com

ⁱⁱ **Francisca Risolene Fernandes Rocha**, ORCID: <https://orcid.org/0000-0002-9017-2142>

Universidade Federal do Ceará, Programa de Pós-graduação em Artes, Prefeitura de Horizonte. Mestre em Artes pelo PPG profissional em Artes da Universidade Federal do Ceará (UFC), Especialista em Alfabetização e Multiletramentos; em Gestão Pedagógica da Escola Básica; e em Língua Portuguesa e Literatura Brasileira, pela Universidade Estadual do Ceará (UECE); Graduada em Pedagogia pela Universidade Estadual Vale do Acaraú (UVA),
Contribuição de autoria: escrita do texto e primeira revisão.
Lattes: <http://lattes.cnpq.br/1700981050573327>
E-mail: profarisolenefernandes@gmail.com

ⁱⁱⁱ **Vitória Chérida Costa Freire**, ORCID: <https://orcid.org/0000-0002-8029-5907>

Prefeitura Municipal de Fortaleza, Secretaria Municipal de Educação (SME)
Licenciada em Pedagogia (2015) pela Universidade Estadual do Ceará (UECE). Possui mestrado (2017) e doutorado (2022) em Educação. Estuda e desempenha pesquisas científicas na área

educacional, principalmente sobre História da Educação no Brasil e no Ceará, Educação de Mulheres, Biografia, Escola Pública e Formação de Professores.

Contribuição de autoria: escrita do texto e segunda revisão final.

Lattes: <http://lattes.cnpq.br/3973477219174231>

E-mail: vitoriacherida91@gmail.com

19

Editora responsável: Genifer Andrade

Especialista *ad hoc*: Victor Hugo de Oliveira Henrique e Antônio Roberto Xavier.

Como citar este artigo (ABNT):

RIBEIRO, Francisco Davi Camilo.; ROCHA, Francisca Risolene Fernandes.; FREIRE, Vitória Chérida Costa. O caminho da Tartaruga: Python e a biblioteca Turtle na educação.

Rev. Pemo, Fortaleza, v. 7, e15701, 2025. Disponível em:
<https://revistas.uece.br/index.php/revpemo/article/view/15701>

Recebido em 11 de junho de 2025.

Aceito em 3 de julho de 2025.

Publicado em 10 de novembro de 2025.