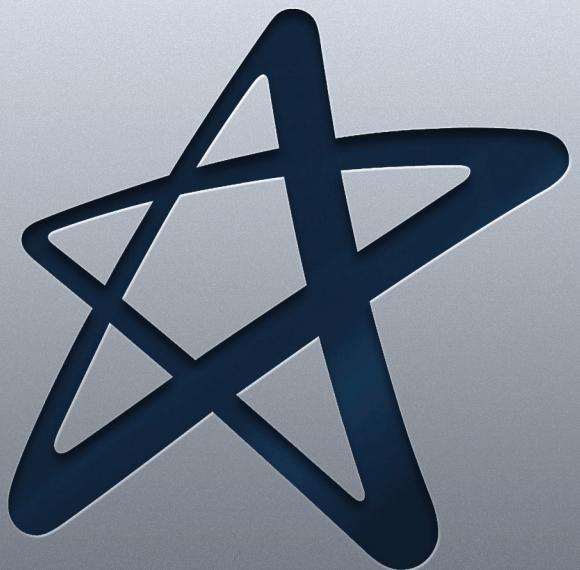


Pensamento Computacional



Cruzeiro do Sul Virtual
Educação a distância

Material Teórico



Decomposição de Problemas e Algoritmos e Procedimentos

Responsável pelo Conteúdo:

Prof. Me. Hugo Fernandes

Revisão Textual:

Prof.^a Me. Natalia Conti

UNIDADE

Decomposição de Problemas e Algoritmos e Procedimentos



- **Decomposição de Problemas;**
- **Algoritmos e Procedimentos.**



OBJETIVO DE APRENDIZADO

- Explorar os conceitos de decomposição de problemas e algoritmos e procedimentos. Essas duas habilidades auxiliam no desenvolvimento da solução de um problema. Ao decompor problemas podemos dar início à busca da solução, generalizando e encontrando padrões para a seguir construir um algoritmo capaz de resolver o problema da forma mais eficiente possível.

Orientações de estudo

Para que o conteúdo desta Disciplina seja bem aproveitado e haja maior aplicabilidade na sua formação acadêmica e atuação profissional, siga algumas recomendações básicas:

Determine um horário fixo para estudar.

Mantenha o foco! Evite se distrair com as redes sociais.

Procure manter contato com seus colegas e tutores para trocar ideias! Isso amplia a aprendizagem.

Seja original! Nunca plágie trabalhos.

Aproveite as indicações de Material Complementar.

Conserve seu material e local de estudos sempre organizados.

Não se esqueça de se alimentar e de se manter hidratado.

Assim:

- ✓ Organize seus estudos de maneira que passem a fazer parte da sua rotina. Por exemplo, você poderá determinar um dia e horário fixos como seu “momento do estudo”;
- ✓ Procure se alimentar e se hidratar quando for estudar; lembre-se de que uma alimentação saudável pode proporcionar melhor aproveitamento do estudo;
- ✓ No material de cada Unidade, há leituras indicadas e, entre elas, artigos científicos, livros, vídeos e sites para aprofundar os conhecimentos adquiridos ao longo da Unidade. Além disso, você também encontrará sugestões de conteúdo extra no item **Material Complementar**, que ampliarão sua interpretação e auxiliarão no pleno entendimento dos temas abordados;
- ✓ Após o contato com o conteúdo proposto, participe dos debates mediados em fóruns de discussão, pois irão auxiliar a verificar o quanto você absorveu de conhecimento, além de propiciar o contato com seus colegas e tutores, o que se apresenta como rico espaço de troca de ideias e de aprendizagem.

Decomposição de Problemas

Decomposição de problemas é a capacidade de dividir uma tarefa em pequenas partes e, em seguida, concentrar-se na resolução de cada uma dessas tarefas menores.

Decomposição de problemas frequentemente nos leva ao reconhecimento e a generalização de padrões e, desse modo, a capacidade de projetar um algoritmo.

Por exemplo, a tarefa de fazer um bolo pode ser decomposta em várias tarefas menores, cada uma das quais pode ser realizada com facilidade. Vejamos uma receita de bolo de chocolate.

Receita de bolo de chocolate

Ingredientes

- 2 xícaras (chá) farinha de trigo
- 2 xícaras (chá) açúcar
- 1 xícara (chá) leite
- 2 ovos
- 4 colheres (sopa) óleo
- 1 pitada de sal
- 1 colher (sopa) fermento em pó
- essência de baunilha a gosto (opcional)

Modo de preparo

- Separe o fermento
- Na batedeira bata os demais ingredientes por aproximadamente 5 minutos
- Retire a tigela da batedeira, junte o fermento e misture cuidadosamente com uma colher
- Despeje a massa do bolo em uma forma de cone untada e enfarinhada
- Leve ao forno preaquecido a 180 °C por aproximadamente 35 minutos ou até que ao espetar um palito ele saia limpo

Nesse cenário o bolo de chocolate é o problema que queremos resolver, para isso utilizando a habilidade de decomposição de problemas, iremos dividir a tarefa (fazer um bolo de chocolate) em pequenas partes e, desse modo, buscar soluções para cada uma dessas partes.

Podemos dividir a tarefa em três partes:

Tabela 1

Decompondo o problema – fazer um bolo de chocolate
1 – Reservar utensílios de cozinha;
2 – Reservar ingredientes;
3 – Realizar procedimentos e passos para mistura de ingredientes.

Nossa tarefa foi decomposta em três tarefas menores. Desse modo, podemos gerenciar melhor nossa tarefa geral, pois trabalhamos com pequenas tarefas por vez. Essa é uma maneira útil de abordar qualquer tarefa ou problema que pareça difícil de resolver de uma só vez.



Trocando ideias...

A habilidade de decomposição de problemas emprega as habilidades de abstração e reconhecimento de padrões. Quando decomponemos um problema precisamos identificar quais são as etapas que podem ser generalizadas (percebemos isso com a abstração) e, também, a existência de padrões (o que tornará essencial para o desenvolvimento de uma solução eficiente).

Um outro exemplo interessante seria a tarefa de desenhar um quadrado. Se pensarmos na decomposição dessa tarefa teríamos basicamente 4 pequenas tarefas:

Tabela 2

Decompondo o problema – desenhar um quadrado
1 – Desenhe uma reta para direita;
2 – Desenhe uma reta para baixo;
3 – Desenhe uma reta para esquerda;
4 – Desenhe uma reta para cima.

O exemplo acima é o resultado da aplicação da estratégia de decompor um problema. Dividimos uma tarefa maior em pequenas tarefas sem nos preocuparmos com a solução, ou seja, sem a preocupação detalhada da solução do problema.

Imagine agora como poderíamos “ensinar” um computador a realizar tais tarefas. Para isso, utilizamos algoritmos.

Algoritmos e Procedimentos

Ao contrário da habilidade de decomposição de problemas, o desenvolvimento de algoritmos e procedimentos é um processo que descreve de forma detalhada o passo a passo para a solução de uma tarefa ou problema. Em posse do problema decomposto podemos criar um **algoritmo** que irá executar a solução para cada pequena tarefa.



Trocando ideias...

Nesse contexto, primeiro decomponemos um problema para depois criarmos um algoritmo.

Você provavelmente conhece o caminho mais rápido para ir de sua casa ao supermercado, por exemplo, vire à esquerda, siga duas quadras, vire à direita. Você pode pensar nisso como um “algoritmo”, uma sequência de procedimentos para

chegar ao seu destino escolhido. Nesse exemplo podemos pensar que existem diversos caminhos possíveis para ir até o supermercado, não é mesmo? Para atingir o mesmo objetivo podemos criar algoritmos diferentes.

Vejamos novamente a tarefa de fazer um bolo de chocolate, a receita de bolo é um algoritmo.

Receita de bolo de chocolate

Ingredientes

- 2 xícaras (chá) farinha de trigo
- 2 xícaras (chá) açúcar
- 1 xícara (chá) leite
- 2 ovos
- 4 colheres (sopa) óleo
- 1 pitada de sal
- 1 colher (sopa) fermento em pó
- essência de baunilha a gosto (opcional)

Modo de preparo

- Separe o fermento
- Na batedeira bata os demais ingredientes por aproximadamente 5 minutos
- Retire a tigela da batedeira, junte o fermento e misture cuidadosamente com uma colher
- Despeje a massa do bolo em uma forma de cone untada e enfarinhada
- Leve ao forno preaquecido a 180 °C por aproximadamente 35 minutos ou até que ao espetar um palito ele saia limpo

Por que consideramos uma receita de bolo um algoritmo?

Porque temos um processo passo a passo que descreve a solução de um problema. Nesse caso, a solução que buscamos é fazer o bolo de chocolate. Como se faz um bolo de chocolate? O algoritmo explica!

O primeiro passo de nosso algoritmo indica que devemos **separar o fermento**.

O segundo passo indica que **na batedeira, bata o restante dos ingredientes por 5 minutos**.

Perceba que existe uma sequência de processos a serem realizados. Caso algum passo não seja devidamente realizado como descrito, o resultado final por certo não será o resultado esperado por quem seguiu a receita/algoritmo. Se você seguir as etapas do algoritmo corretamente, você chegará a uma solução correta.

Voltemos ao exemplo da tarefa de desenhar um quadrado. Já decomponemos o problema:

Tabela 3

Decompondo o problema – desenhar um quadrado
1 – Desenhe uma reta para direita; 2 – Desenhe uma reta para baixo; 3 – Desenhe uma reta para esquerda; 4 – Desenhe uma reta para cima.

Porém, para resolução do problema, devemos criar um algoritmo.

Tabela 4

Algoritmo – desenhar um quadrado
1 – Posicione a caneta na superfície; 2 – Percorra 50 unidades de medida e direcione a caneta 90 graus; 3 – Percorra 50 unidades de medida e direcione a caneta 90 graus; 4 – Percorra 50 unidades de medida e direcione a caneta 90 graus; 5 – Percorra 50 unidades de medida e direcione a caneta 90 graus.

O algoritmo acima descreve detalhadamente o passo a passo para a solução de nosso problema: desenhar um quadrado.

Os cientistas da computação se esforçam para encontrar os algoritmos mais eficazes e eficientes, ou seja, aqueles que resolvem um problema no tempo mais rápido, usando o mínimo de recursos (memória ou tempo) ou da maneira mais eficaz (obtendo o correto ou o mais próximo da resposta correta).

Agora que desenvolvemos um algoritmo podemos facilmente implementá-lo, para o nosso exemplo, iremos utilizar o Scrtach.



O que é Algoritmo? Disponível em: <https://bit.ly/2JiD2zn>.

Implementando o algoritmo no Scratch



Acesse a plataforma Scratch por meio do link: <https://bit.ly/1A9CAq4>.

Comece arrastando para a área de codificação um comando “quando for clicado” conforme podemos observar na figura 3.

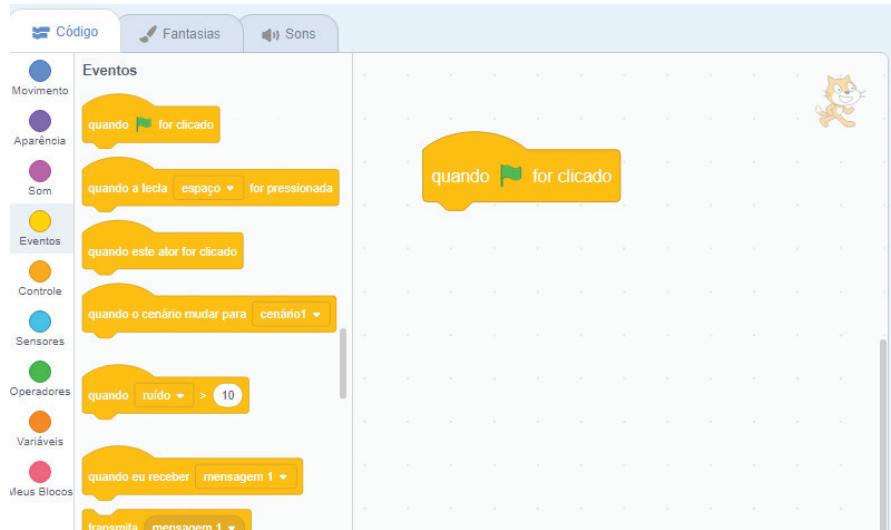


Figura 3 – Arrastando comando “quando for clicado”

Fonte: Scratch

Agora devemos adicionar o bloco Caneta. Para isso, na área de seleção de abas, no canto inferior à esquerda, clique no botão Adicionar uma extensão.

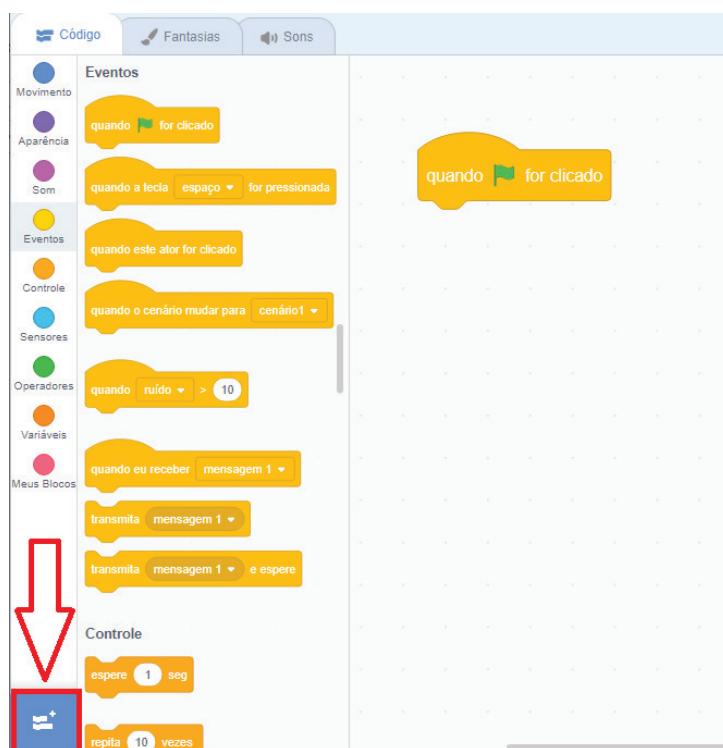


Figura 4 – Adicionando uma extensão

Fonte: Scratch

Na tela seguinte selecione a extensão Caneta.

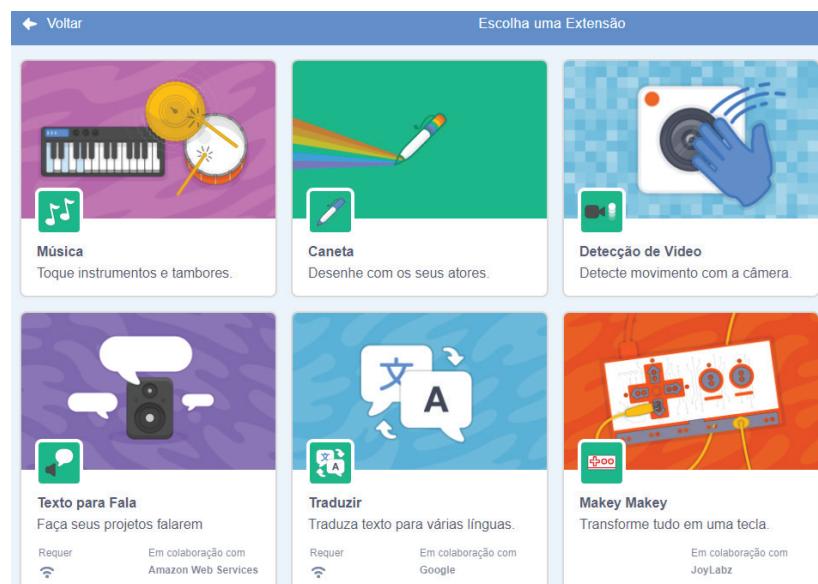


Figura 5 – Extensão caneta

Fonte: Scratch

Esse procedimento faz com que a aba Caneta seja adicionado ao Scratch. Agora, na aba Caneta, localize e arraste o comando “Use a caneta”.

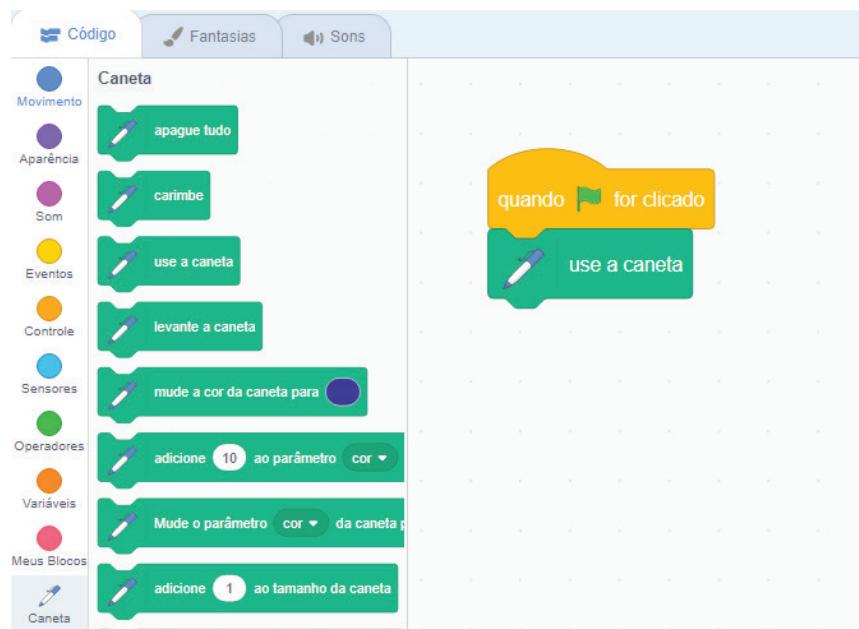


Figura 6 – Extensão caneta

Fonte: Scratch

Até o momento, nosso programa já está implementando o passo 1 de nosso algoritmo.

Tabela 5

Algoritmo – desenhar um quadrado	
1 – Posicione a caneta na superfície;	
2 – Percorra 50 unidades de medida e direcione a caneta 90 graus;	
3 – Percorra 50 unidades de medida e direcione a caneta 90 graus;	
4 – Percorra 50 unidades de medida e direcione a caneta 90 graus;	
5 – Percorra 50 unidades de medida e direcione a caneta 90 graus.	

Precisamos agora implementar o restante. Para isso, na aba Movimento localize e arraste os comandos “mova” e “gire graus” e os posicione conforme a imagem 7.

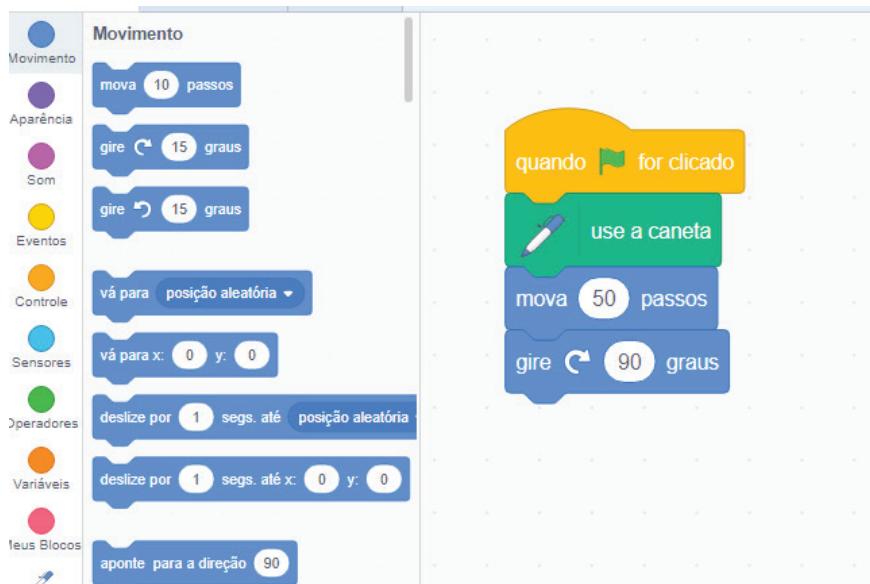


Figura 7. Comandos de movimento

Fonte: Scratch

Com isso implementamos o passo 2 de nosso algoritmo.

Tabela 6

Algoritmo – desenhar um quadrado	
1 – Posicione a caneta na superfície;	
2 – Percorra 50 unidades de medida e direcione a caneta 90 graus;	
3 – Percorra 50 unidades de medida e direcione a caneta 90 graus;	
4 – Percorra 50 unidades de medida e direcione a caneta 90 graus;	
5 – Percorra 50 unidades de medida e direcione a caneta 90 graus.	

Note que o passo 3, 4 e 5 de nosso algoritmo são iguais ao nosso passo 2. Sendo assim, repita o processo, ou seja, arraste os comandos “mova” e “gire graus” mais 3 vezes e os posicione conforme a imagem 8.

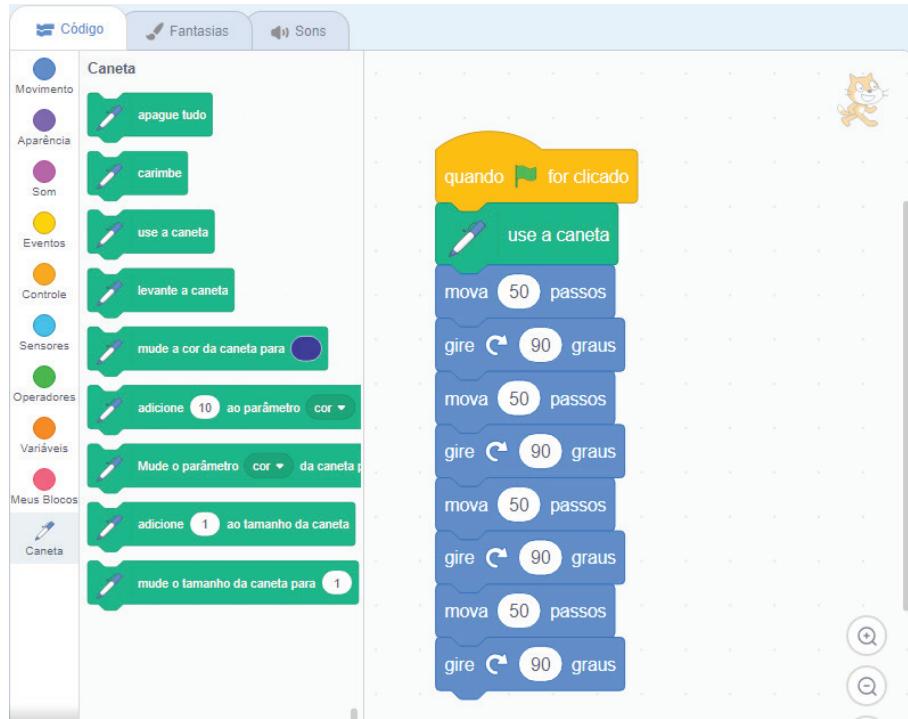


Figura 8 – Comandos de movimento

Fonte: Scratch

Testando nosso programa podemos perceber que ele realiza a tarefa de desenhar um quadrado.

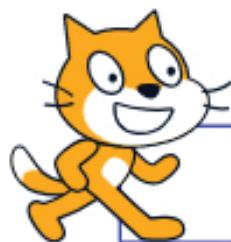


Figura 9 – Testando o programa

Fonte: Scratch

Melhorando nosso código

Nosso programa já executa o que é esperado, ou seja, desenha um quadrado, porém, ainda podemos melhorar nosso programa utilizando a estratégia que aprendemos na aula sobre **reconhecimento de padrões**. Por exemplo, analisando o código, percebemos que os comandos “mova” e “gire graus” se repetem, sendo assim, podemos melhorar nosso programa utilizando um laço repita “4” vezes. Desse modo, nosso programa em comparação ao original teria uma codificação mais limpa e intuitiva.

Para tanto, remova todos os blocos abaixo do comando “use a caneta”, conforme a figura 10.



Figura 10 – Melhorando nosso código

Fonte: Scratch

Em seguida, na aba Controle, localize e arraste um bloco “repita vezes” e configure a quantidade de voltas de nosso laço para 4 vezes, ficando conforme a figura 11.



Figura 11 – Inserindo um bloco repita

Fonte: Scratch

Agora, arraste os blocos “mova passos” e “gire graus” para dentro do bloco repita. Para o bloco mova configure o valor de 50 passos. Para o bloco gire graus configure o valor para 90 graus. Podemos ver essas configurações na imagem 12.



Figura 12 – Configurando blocos mova e gire graus

Fonte: Scratch

Por fim, façamos o teste novamente de nosso programa.

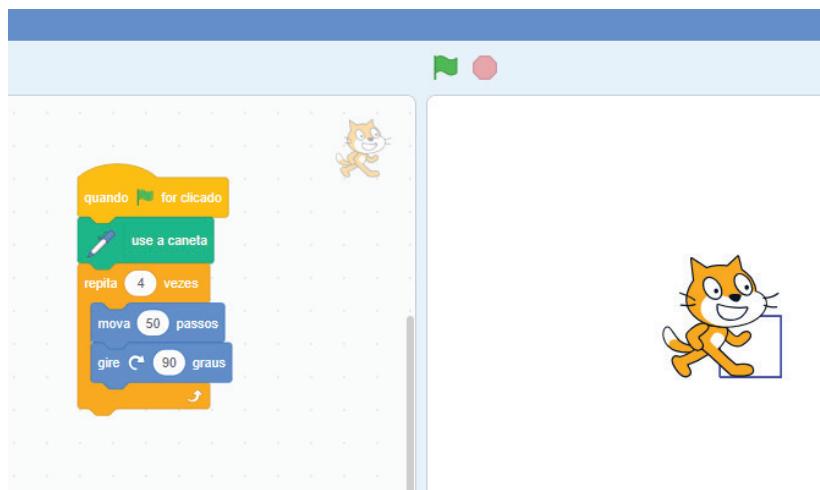


Figura 13 – Testando o programa

Fonte: Scratch

Podemos perceber que nosso programa executa a tarefa de acordo com o objetivo, ou seja, desenhou um quadrado, porém, agora nosso programa possui um código mais intuitivo e de fácil alteração, uma vez que o trecho de código responsável por desenhar um quadrado na tela se resume em 4 comandos e não em 9, como no código original.



Escreva códigos para pessoas e não para máquinas: <https://bit.ly/2UUXqfj>.

Em conclusão, exploramos os conceitos de decomposição de problemas e algoritmos e procedimentos, dois dos mais poderosos conceitos do Pensamento Computacional. Uma vez conhecidos e utilizados, esses conceitos são extremamente importantes para o desenvolvimento e criação de soluções para tarefas e problemas, sejam elas com objetivo de implementação utilizando computadores ou procedimentos despluggedos.

Material Complementar

Indicações para saber mais sobre os assuntos abordados nesta Unidade:

▶ Vídeos

O que é e como funcionam os algoritmos?

https://youtu.be/Xo1V_JL1yAg

📄 Leitura

Afinal, o que é um algoritmo e o que isso tem a ver com computação?

<https://bit.ly/2PJQpwI>

Na verdade, o que [...] é exatamente um algoritmo?

<https://bit.ly/2Cpdj8b>

O que é um algoritmo, e por que você deve se importar com isso?

<https://bit.ly/2Ya01Eb>

Referências

BBC LEARNING, B. *What is computational thinking?*, 2018. Disponível em: <<http://www.bbc.co.uk/education/guides/zp92mp3/revision>>. Acesso em: 01/02/2018.

BELL, T.; WITTEN, I. H.; FELLOWS, M. *Computer Science Unplugged: An enrichment and extension programme for primary-aged students*. 2015.

WING, J. M. *Computational thinking*. *Communications of the ACM*, v. 49, n. 3, p. 33–35, 2006.



Cruzeiro do Sul
Educacional