

The background features a collection of 3D blocks in various colors (purple, blue, yellow, green, orange, pink, grey) arranged in a staggered, ascending pattern from the bottom left towards the top right. A thin, multi-colored curved line (red, blue, green) starts near the bottom left and arcs upwards towards the top right, passing behind the blocks.

# Leitura complementar

# Algoritmo

- ✓ Algoritmo;
- ✓ Elementos de um algoritmo;
- ✓ Algoritmo com o comando SE;
- ✓ Algoritmo com o comando CASO;
- ✓ Algoritmo com o comando ENQUANTO.

# Algoritmo

**Algoritmo** é a descrição sequencial ordenada dos passos que devem ser executados, de forma lógica e clara, com a finalidade de facilitar a resolução de um problema.

Você viu anteriormente o passo a passo da rotina do início do dia de uma pessoa. Então, é o algoritmo que ajuda a resolver o que fazer desde a hora de acordar até chegar ao local de trabalho.

A seguir temos um algoritmo, com numeração nas linhas, de **como fritar um ovo**.

Detalhes: Um ovo, uma lata de óleo, um saleiro, um prato, uma colher e uma caixa de fósforos estão ao lado do fogão, em cima de uma pia. Tem gás e a frigideira já está em cima de uma boca do fogão.

#### Início

1. Acender o fogo
  2. Colocar óleo na frigideira
  3. O óleo está quente?
    - 3.1. Se sim: “Próximo passo”
    - 3.2. Se não: “Esperar o óleo esquentar” e “Vá para o passo 3”
  4. Quebrar o ovo
  5. O ovo está com uma boa consistência?
    - 5.1. Se sim: “Próximo passo”
    - 5.2. Se não: “Vá para o passo 10”
  6. Colocar o ovo na frigideira
  7. Colocar uma pitada de sal
  8. O ovo está frito?
    - 8.1. Se sim: “Próximo passo”
    - 8.2. Se não: “Esperar fritar” e “Vá para o passo 8”
  9. Pegar o ovo com a escumadeira e colocar em um prato
  10. Desligar o fogo
- Fim

A seguir temos um algoritmo de **como trocar um pneu**.

Detalhes: O carro está estacionado num lugar seguro e quem vai trocar o pneu já está ao lado do porta-malas com as chaves do carro.

#### Início

1. Abrir o porta-malas do carro
2. Tem estepe?
  - 2.1. Se sim: Retirar o estepe
  - 2.2. Se não: Vá para o passo 17
3. Tem ferramentas?
  - 3.1. Se sim: Pegar as ferramentas
  - 3.2. Se não: Vá para o passo 16
4. Desapertar os parafusos da roda
5. Já desapertou todos os parafusos?
  - 5.1. Se sim: Vá para o passo 6
  - 5.2. Se não: Vá para o passo 4
6. Levantar o carro com o macaco
7. Levantou o suficiente?
  - 7.1. Se sim: Próximo passo
  - 7.2. Se não: Vá para o passo 6
8. Retirar os parafusos
9. Retirar a roda
10. Colocar o estepe
11. Recolocar os parafusos
12. Abaixar o carro
13. Apertar os parafusos
14. Já apertou todos os parafusos?
  - 14.1. Se sim: Vá para o passo 15
  - 14.2. Se não: Vá para o passo 13
15. Guardar as ferramentas
16. Guardar o pneu
17. Fechar o porta-malas

#### Fim

Os algoritmos de ações cotidianas geralmente permitem que uma pessoa o execute mesmo que ela nunca tenha feito tal coisa.

Uma receita de um bolo é um exemplo. Você pode fazer o bolo se seguir corretamente a receita. Agora, se a receita não estiver detalhada o suficiente, já não se pode garantir o resultado.

Um algoritmo detalhado mostra ações importantes como **Desligar o fogo**, no primeiro algoritmo, ou **Guardar o pneu**, no segundo algoritmo.

# Elementos de um algoritmo

Veremos, a seguir, os elementos que compõem um algoritmo.

## Ação

- Abrir o porta-malas do carro
- Retirar os parafusos
- Retirar a roda
- Colocar o estepe
- Recolocar os parafusos
- Abaixar o carro
- Guardar as ferramentas
- Guardar o pneu
- Fechar o porta-malas

## Decisão

**Tem estepe?**


**Se sim: Retirar o estepe**

**Se não: Vá para o passo 17**

**Tem ferramentas?**

**Se sim: Pegar ferramentas**

**Se não: Vá para o passo 16**



**Observação:** Note que, numa decisão, existem duas respostas ou caminhos a seguir: o lado verdadeiro e o lado falso da indagação.


## Laço ou Loop

**4. Desapertar os parafusos da roda**

**5. Já desapertou todos os parafusos?**

**5.1. Se sim: Vá para o passo 6**

**5.2. Se não: Vá para o passo 4**




**Observação:** Note que, se não foram desapertados todos os parafusos, a execução do programa volta para a linha 4, fazendo um loop, que são trechos de uma lógica que podem ser executados várias vezes.

**6. Levantar o carro com o macaco**

**7. Levantou o suficiente?**

**7.1. Se sim: Próximo passo**

**7.2. Se não: Vá para o passo 6**




**Observação:** Note que, se o carro não foi suficientemente levantado, a execução do programa volta para a linha 6, fazendo um loop. Observe também que, em caso positivo, a execução do programa foi para o **Próximo passo**, que seria a mesma coisa que **Ir para o passo 8**, que é o próximo passo.

**13. Apertar os parafusos**

**14. Já apertou todos os parafusos?**

**14.1. Se sim: Vá para o passo 15**

**14.2. Se não: Vá para o passo 13**



**Observação:** Note que, se não foram apertados todos os parafusos, a execução do programa volta para a linha 13, fazendo um loop.

# Algoritmo com o comando SE encadeado

Já vimos que o comando **SE** tem duas respostas, **SE SIM** e **SE NÃO**, mas pode ser que você precise de mais alternativas como resposta.

A solução é **encadear** o comando **SE**, colocando um dentro do outro.

- **Exemplo: Menu principal de um programa**

**Início**

**1. Digite um número de 1 a 6**

**2. O número digitado é 1?**

**2.1. Se sim: Abrir o programa de Inclusão**

**2.2. Se não: O número digitado é 2?**

**2.2.1. Se sim: Abrir o programa de Exclusão**

**2.2.2. Se não: O número digitado é 3?**

**2.2.2.1. Se sim: Abrir o programa de Consulta**

**2.2.2.2. Se não: O número digitado é 4?**

**2.2.2.2.1. Se sim: Abrir o programa de Alteração**

**2.2.2.2.2. Se não: O número digitado é 5?**

**2.2.2.2.2.1. Se sim: Abrir o programa de Impressão**

**2.2.2.2.2.2. Se não: O número digitado é 6?**

**2.2.2.2.2.2.1. Se sim: Fechar Menu**

**2.2.2.2.2.2.2. Se não: Exibir “Número Inválido” e**

**“Ir para o passo 1”**

**Fim**

# Algoritmo com o comando CASO

Já vimos que o comando **SE** tem duas respostas, **SE SIM** e **SE NÃO**. O comando **CASO** pode ter quantas respostas você precisar. Ele é usado como alternativa ao comando **SE** encadeado.



**Observação:** Nem todas as linguagens tem o comando **CASO**, e somente no comando **SE** existe **SE SIM** e **SE NÃO**.

- Exemplo: Menu principal de um programa

**Início**

**1. Digite um número de 1 a 6**

**2. Caso**

**2.1. Caso digitou 1: Abrir o programa de Inclusão**

**2.2. Caso digitou 2: Abrir o programa de Exclusão**

**2.3. Caso digitou 3: Abrir o programa de Consulta**

**2.4. Caso digitou 4: Abrir o programa de Alteração**


**2.5. Caso digitou 5: Abrir o programa de Impressão**

**2.6. Caso digitou 6: Fechar Menu**

**2.7. Caso contrário: Exibir “Número Inválido” e  
“Ir para o passo 1”**

**3. Fim do Caso**

**Fim**



**Observação:** Note que na linha 4, o comando **Fim do Caso** serve pra indicar onde termina o comando **CASO**. Depois o programa continua normalmente a partir da próxima linha, que, neste caso, é o comando **Fim**, que serve para identificar o final do algoritmo.



# Algoritmo com o comando ENQUANTO

O comando **ENQUANTO**, que em inglês significa **WHILE**, serve para que uma parte do programa seja repetida enquanto uma situação for satisfeita. O loop que repete as linhas de programação só é interrompido quando a condição que o faz repetir os comandos não for mais verdadeira.

Exemplo: Vamos supor que exista uma caixa com 30 bolas, sendo 10 verdes, 10 amarelas e 10 azuis. O objetivo é retirar bolas da caixa, sem olhar, até ter certeza de que saíram 3 bolas de cor azul. Para facilitar utilizaremos contadores de bolas, que inicialmente têm o valor zero (0), porque ainda nenhuma bola foi retirada.

#### Início

#### 1. Enquanto o contador de bolas azuis for menor que 03

##### 1.1 Retirar uma bola

##### 1.2 Caso

1.2.1 Caso a bola seja verde: Somar 1 no contador de bolas verdes

1.2.2 Caso a bola seja amarela: Somar 1 no contador de bolas amarelas

1.2.3 Caso contrário: Somar 1 no contador de bolas azuis

##### 1.3 Fim do Caso

#### 2. Fim do Enquanto

#### 3. Exibir a mensagem “Três bolas azuis foram retiradas da caixa”.

Fim

Note que, dentro do comando **ENQUANTO**, foram colocados os comandos de retirar uma bola e o comando **CASO** que verifica a cor das bolas somando 1 no contador da respectiva cor. Após o comando **Caso contrário: Somar 1 no contador de bolas azuis**, a execução volta diretamente para o comando **Enquanto o contador de bolas azuis for menor que 03**.

Após esta verificação, caso o contador de bolas azuis não seja então menor que 3, a execução vai para **Exibir a mensagem Três bolas azuis foram retiradas da caixa**. Caso contrário, a execução entra novamente no loop e retira mais uma bola.