

The background features a collection of 3D cubes in various colors (purple, blue, yellow, green, orange, pink, grey) arranged in a scattered, overlapping pattern. Some cubes are solid, while others have a square hole in the center. Thin, curved lines in blue, red, and green connect some of the cubes, suggesting a network or flow. The overall aesthetic is clean and modern, with a light grey gradient background.

Mãos à obra! - Respostas

Algoritmo

Exercício 1

A proposta do exercício é construir um algoritmo que simule uma **ligação telefônica**, considerando que possuímos um **telefone fixo** e que ele esteja funcionando corretamente. O resultado é um algoritmo parecido com este:

```
1.Início
  2.Tirar o fone do gancho
  3.Verificar se tem linha
    Se sim: Vá para o passo 4
    Se não: Colocar o telefone no gancho
            Vá para o passo 2
  4.Teclar o número desejado
  5.Verificar se está chamando
    Se sim: Aguardar
            Vá para o passo 6
    Se não: Colocar o telefone no gancho
            Vá para o passo 2
  6.Atenderam?
    Se sim: Conversar
            Vá para o passo 7
    Se não: Tentar mais tarde
  7.Colocar o fone no gancho
  8.Fim
```

Veja como criá-lo:

Primeiro vamos analisar o problema que temos em mãos e, depois, dividi-lo em uma sequência de passos.

Quando vamos fazer uma ligação, a primeira coisa a ser feita é tirar o fone do gancho. Depois verificamos se a linha está disponível. Repare que esta será uma estrutura de decisão como as que já vimos durante a aula, afinal há dois caminhos possíveis para o nosso algoritmo: ou a linha pode estar disponível, que representa o fluxo verdadeiro, ou pode estar sem sinal, que representa o fluxo falso. Se a linha estiver disponível, vamos para o próximo passo. Se não, colocamos o telefone no gancho e voltamos para o início do algoritmo.

```
1.Início
  2.Tirar o fone do gancho
  3.verificar se tem linha
    Se sim: Vá para o passo 4
    Se não: Colocar o telefone no gancho
            Vá para o passo 2
```

Depois disso, **digitamos o número desejado** e verificamos se está chamando. Se sim, **basta aguardar na linha**. Se não, **colocamos o telefone no gancho e voltamos para o início do algoritmo**.

```
4.Teclar o número desejado
5.Verificar se está chamando
  Se sim: Aguardar
          Vá para o passo 6
  Se não: Colocar o telefone no gancho
          Vá para o passo 2
```

Com a confirmação de que o telefone está chamando, vamos verificar logo em seguida se seremos atendidos ou não. Temos aqui outra estrutura de decisão. **Se formos atendidos, conversamos normalmente com a pessoa do outro lado da linha. Se não, colocamos o telefone no gancho e tentamos mais tarde.** A ação “**Tentar mais tarde**”, nesse caso, finaliza o algoritmo, já que, se ninguém nos atender, não vamos voltar para o início do algoritmo.

```
6. Atenderam?  
    Se sim: Conversar  
           Vá para o passo 7  
  
    Se não: Tentar mais tarde  
7. Colocar o fone no gancho  
8. Fim
```

Depois de conversar, **colocamos o fone no gancho**, chegando ao fim do nosso algoritmo.

Quando estamos codificando e realizando os exercícios, é uma boa prática facilitar ao máximo o entendimento do algoritmo. Um fator que contribui muito para isso é a **indentação** do código. Veja o algoritmo que acabamos de criar. Sempre que temos uma decisão, aplicamos um recuo nos blocos de instruções que representam o fluxo que o algoritmo poderá seguir. Fazendo isso, é só bater o olho que conseguimos saber que estes dois trechos, **Se sim** e **Se não**, estão relacionados à decisão acima.

```
4. Teclar o número desejado  
5. verificar se está chamando  
    Se sim: Aguardar  
           Vá para o passo 6  
    Se não: Colocar o telefone no gancho  
           Vá para o passo 2
```