

# Pensando Computacionalmente

Wladimir Araújo Tavares<sup>1</sup>

<sup>1</sup>Universidade Federal do Ceará - Campus de Quixadá

- 1 Introdução
- 2 Decomposição
- 3 Reconhecimento de Padrão
- 4 Abstração
- 5 Algoritmo
- 6 Sequência Didática Sugerida

# O que é o Pensamento Computacional?

## Pensamento Computacional

O pensamento computacional é a habilidade de resolver um problema descrevendo a sua solução de maneira que o computador possa executar.

# Importância do Pensamento Computacional

O relatório do Fórum Mundial apresenta as **15 habilidades mais importantes** no mercado de trabalho até 2025 <sup>1</sup>:

1.	<b>Pensamento analítico e inovação</b>
2.	<b>Aprendizagem ativa</b> e estratégias de aprendizado
3.	<b>Resolução de problemas</b>
4.	<b>Pensamento crítico</b>
5.	<b>Criatividade</b>
6.	Liderança
7.	Uso, monitoramento e controle de tecnologias
8.	<b>Programação</b>
9.	Resiliência, tolerância ao estresse e flexibilidade
10.	<b>Raciocínio lógico</b>
11.	Inteligência emocional
12.	Experiência do usuário
13.	Ser orientado a servir o cliente (foco no cliente)
14.	Análise e avaliação de sistemas
15.	Persuasão e negociação

<sup>1</sup>[https://www3.weforum.org/docs/WEF\\_Future\\_of\\_Jobs\\_2020.pdf](https://www3.weforum.org/docs/WEF_Future_of_Jobs_2020.pdf)

# Técnicas principais do Pensamento Computacional

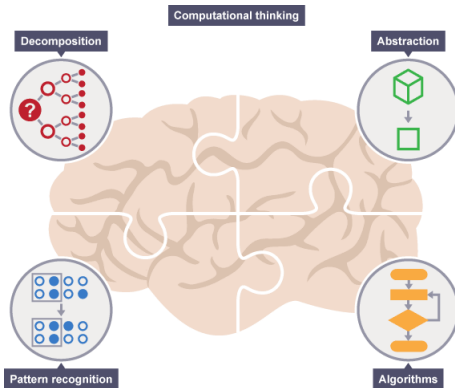


Figura: Fonte: Introduction to computational thinking <sup>2</sup>

<sup>2</sup><https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1>

# Resolvendo problemas complexos

- Divida um problema complexo em uma série de problemas pequenos e mais gerenciáveis (decomposição).
- Encontre similaridades entre os problemas menores (reconhecimento de padrão).
- Foque apenas nos detalhes importantes e ignore informações irrelevantes (abstração).
- Projete a solução dos problemas menores utilizando regras simples.

- 1 Introdução
- 2 Decomposição
- 3 Reconhecimento de Padrão
- 4 Abstração
- 5 Algoritmo
- 6 Sequência Didática Sugerida

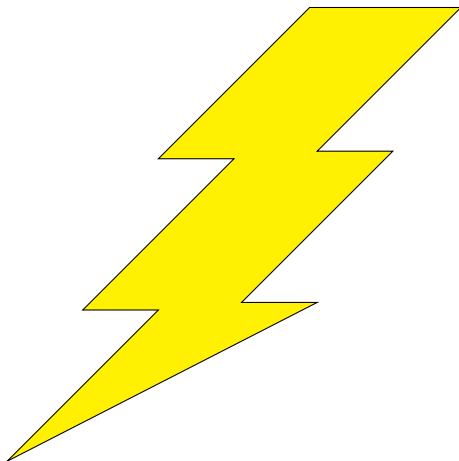
# Decomposição

- A decomposição é o primeiro passo no processo de resolução utilizando o pensamento computacional.
- Um problema difícil pode ser dividido em partes mais simples que podem ser resolvido diretamente.



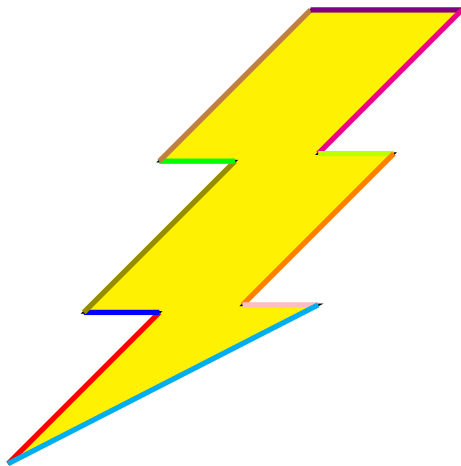
# Decomposição

Considere a tarefa de desenhar a seguinte figura:



# Decomposição

O problema pode ser decomposto em desenhar 11 pequenas partes.



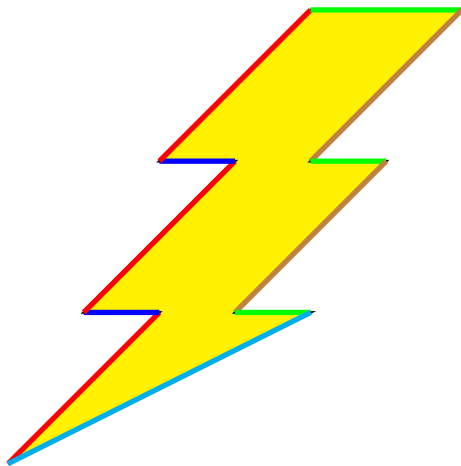
- 1 Introdução
- 2 Decomposição
- 3 Reconhecimento de Padrão**
- 4 Abstração
- 5 Algoritmo
- 6 Sequência Didática Sugerida

# Reconhecimento de Padrão

- Os padrões são características e similaridades que são compartilhadas entre os subproblemas encontrados pela decomposição.
- Essas similaridades podem ajudar a resolver a tarefa mais complexa.

# Reconhecimento de Padrão

Podemos separar em 5 padrões diferentes.



- 1 Introdução
- 2 Decomposição
- 3 Reconhecimento de Padrão
- 4 Abstração**
- 5 Algoritmo
- 6 Sequência Didática Sugerida

# Abstração

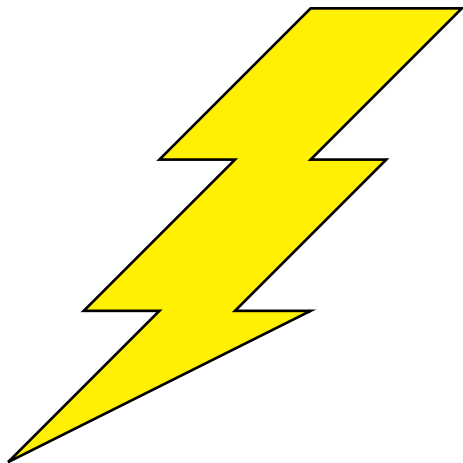
- A abstração pode ser entendida como a definição das características mais importantes e a filtragem dos detalhes e características que não precisamos.
- Características gerais e detalhes para assar um bolo <sup>3</sup>

Características Gerais	Detalhes
Precisamos saber que um bolo tem ingredientes	Não precisamos saber quais são esses ingredientes
Precisamos saber que cada ingrediente tem uma quantidade especificada	Não precisamos saber qual é essa quantidade
Precisamos saber que cada bolo precisa de um tempo específico para assar	Não precisamos saber quanto tempo é o tempo

<sup>3</sup><https://www.bbc.co.uk/bitesize/guides/zttrcdm/revision/2>

# Abstração

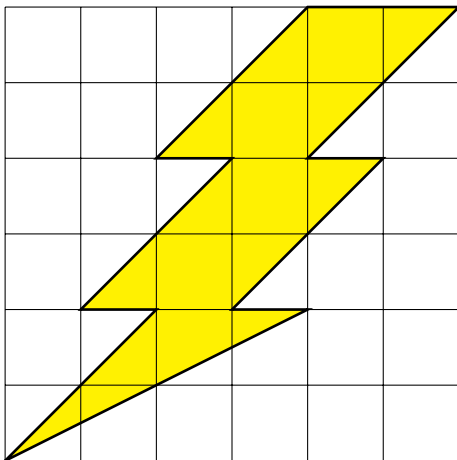
A característica mais importante de cada subproblema é que todos são retas que podem ser definidas por dois pontos.





# Abstração

Note que a abstração é bastante importante para conseguirmos resolver o problema com o auxílio do computador. Vamos colocar a nossa imagem no plano cartesiano.



- 1 Introdução
- 2 Decomposição
- 3 Reconhecimento de Padrão
- 4 Abstração
- 5 Algoritmo**
- 6 Sequência Didática Sugerida

# Algoritmo

- Um algoritmo é um plano passo a passo para resolver um problema.
- Em um algoritmo, cada instrução é identificada e a ordem em que devem ser executadas é definida.
- Os algoritmos são frequentemente usados como ponto de partida para a criação de um programa de computador e, às vezes, são escritos como um fluxograma ou em pseudocódigo.

---

**Algorithm 1** Desenhando Figura

---

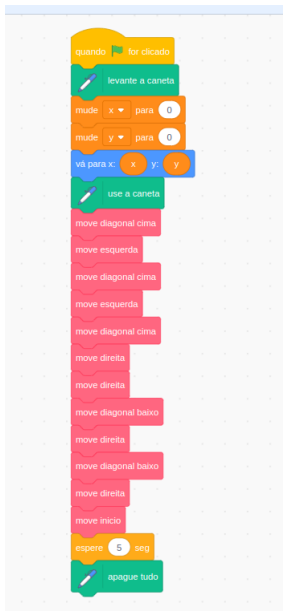
Desenhe um traço diagonal 2 unidades para cima;  
Desenhe um traço para a esquerda de 1 unidade;  
Desenhe uma traço diagonal 2 unidades para cima;  
Desenhe um traço para a esquerda de 1 unidade;  
Desenhe uma traço diagonal 2 unidades para cima;  
Desenhe um traço a direita 2 unidade;  
Desenhe uma traço diagonal 2 unidades para baixo;  
Desenhe um traço para a direita de 1 unidade;  
Desenhe uma traço diagonal 2 unidades para baixo;  
Desenhe um traço para a direita de 1 unidade;  
Desenhe uma traço para o ponto inicial;

---

# Algoritmo

- Note que o algoritmo apresenta as instruções e a ordem em que as instruções devem ser executadas.
- Lembrando que o algoritmo é independente da linguagem. A implementação do algoritmo em uma linguagem específica depende dos recursos disponíveis na linguagem
- Vamos mostrar como esse algoritmo usando a linguagem Scratch

# Algoritmo utilizando a linguagem Scratch



# Algoritmo utilizando a linguagem Scratch



- 1 Introdução
- 2 Decomposição
- 3 Reconhecimento de Padrão
- 4 Abstração
- 5 Algoritmo
- 6 Sequência Didática Sugerida**



# Sequência Didática

- Uma sequência didática é um encadeamento de passos ligados entre si para tornar mais eficiente o processo de aprendizagem.
- As sequências didáticas são planejadas e desenvolvidas para a realização de determinados objetivos educacionais.
- Vamos considerar que a sequência didática deve apresentar os objetivos, público-alvo, conteúdo, tempo para execução da atividade, Recursos Necessários e o detalhamento da atividade em passos.

# Sequência Didática

- **Objetivos:** Desenvolver a criatividade, resolução de problemas e o pensamento computacional.
- **Público-alvo:** Alunos a partir do primeiro ano do Ensino Médio.
- **Conteúdo:** Decomposição, Reconhecimento de Padrão e Algoritmo.
- **Tempo:** 30 minutos
- **Recursos:** Quadro e pincel
- **Recursos opcionais:** Computador com o Scratch instalado.

# Sequência Didática

- **Passo 1:** Apresentação da atividade

Nesse passo, o professor apresenta a figura que será desenhada. O professor pode separar a sala de aula em equipes. Cada equipe deve criar uma descrição de uma sequência de passos que será executada por um outro aluno no quadro da outra equipe que deve ficar com os olhos fechados ou uma venda.

- **Passo 2:** Execução da sequência de instrução

Nesse passo, o professor vai avaliar a execução da sequência de instruções. O aluno que está executando deve seguir as instruções que foram criadas pela outra equipe.

# Sequência Didática

- **Passo 3:** Avaliação da sequência de instrução  
Nesse passo, os alunos devem avaliar todos os algoritmos apresentados por todas as equipes. Destacando os pontos fracos e fortes de cada solução.
- **Passo 4:** Utilizando o scratch  
Nesse passo, os alunos serão desafiados a tentarem transpor seu algoritmo para a linguagem Scratch.
- **Passo 5:** Avaliação  
A avaliação deve levar em conta a fase de construção e execução da sequência de instruções.