# COMPUTABILIDADE E COMPLEXIDADE DE ALGORITMOS

## INTRODUÇÃO

- **Análise de Algoritmos**

- Pode-se determinar se um algoritmo é o mais eficiente utilizando duas abordagens:
  - **Análise Empírica:** comparação entre os programas;
  - **Análise Matemática:** estudo das propriedades dos algoritmos;

# INTRODUÇÃO

- **Análise de Algoritmos**

- **Análise empírica:**
  - Avalia o custo de um algoritmo já implementado e em execução;
  - Logo, é analisado o programa!
  - Considera custos não aparentes;
  - Utilizado para comparar computadores e linguagens;
  - Resultado pode ser injusto.

# INTRODUÇÃO

- **Análise de Algoritmos**

- **Análise matemática:**
  - Permite um estudo formal do algoritmo;
  - Considera somente os custos dominantes do algoritmo;
  - A medição do tempo (custo) é feita de maneira independente do hardware ou da linguagem de programação utilizada;
  - Permite compreender o comportamento de um algoritmo à medida que a instância do problema (conjunto de dados de entrada n) cresce.

# INTRODUÇÃO

- **Complexidade**

- Considere 5 algoritmos A1, A2, A3, A4 e A5 para resolver um mesmo problema, de complexidades diferentes. (Vamos partir do pressuposto que cada operação leva 1 milissegundo para ser efetuada).
- T(n) é a complexidade, ou seja, o número de operações que o algoritmo efetua para n entradas.

| **n** | **A1**<br>**T(n) = n** | **A2**<br>**T(n) = nlogn** | **A3**<br>**T(n) = $n^2$** | **A4**<br>**T(n) = $n^3$** | **A5**<br>**T(n) = $2^n$** |
|---|---|---|---|---|---|
| 16 | 0,016s | 0,064s | 0,256s | 4s | 1m4s |
| 32 | 0,032s | 0,16s | 1s | 33s | 46 dias |
| 512 | 0,512s | 9s | 4m22s | 1 dia 13 h | $10^{137}$ séculos |

# INTRODUÇÃO

- **Complexidade**

- Análise do impacto de um aumento de velocidade sobre alguns algoritmos:

- **Algoritmo linear:** o tempo de execução é proporcional ao tamanho da entrada. Um problema com tamanho máximo $x_1$ é resolvido em um computador em um tempo $t$. Neste caso, um computador dez vezes mais rápido resolverá no mesmo tempo o correspondente a $10x_1$.

# INTRODUÇÃO

- **Complexidade**

- Análise do impacto de um aumento de velocidade sobre alguns algoritmos:

- **Algoritmo quadrático:** o tempo de execução corresponde a $n^2$ para o tamanho da entrada $n$. Um determinado problema chamado $x_3$ pode ser resolvido em um tempo $t$, ou seja, $x_3^2 = t$. Considerando o computador 10 vezes mais rápido, isto é, um tempo de $10t$, o tamanho do problema resolvido $(y)$ será:
- $y^2 = 10t \therefore y^2 = 10(x_3)^2 \therefore y = x_3\sqrt{10} \therefore y \cong 3{,}16x_3$

# INTRODUÇÃO

- **Complexidade**

- Análise do impacto de um aumento de velocidade sobre alguns algoritmos:

- **Algoritmo exponencial:** o tempo de execução consiste em $2^n$ para o tamanho da entrada $n$. Um determinado problema chamado $x_5$ pode ser resolvido em um tempo $t$, ou seja, $2^{x_5} = t$. O computador dez vezes mais rápido terá o volume estabelecido por $(y)$:

- $2^{x_5} = t \; e \; 2^y = 10t \therefore 2^y = 10.2^{x_5} \therefore y = log_2 10 + x_5 \therefore y \cong x_5 + 3{,}3$

# INTRODUÇÃO

- **Complexidade**

- Quais são os valores de tamanho máximo para $n^3$ e $3^n$, considerando os exemplos anteriores ?

# COMPUTABILIDADE E COMPLEXIDADE DE ALGORITMOS

- **OBRIGADO !!**