RENAN GABRIEL

Complexidade de algoritmos

A complexidade de algoritmos, funciona como um medidor de velocidade e quanto de memória ele consome, variando de acordo com o valor do input dele.

* Big O Notation
* É uma notação que indica a complexidade dos algoritmos:

principais funções de complexidade, com ênfase na complexidade de tempo:

1. **Constantes O(1)**

**- O número de operações não muda, mesmo com o valor do input**

1. **Complexidade Linear O(n)**

**- O número de operações só vai crescendo, mesmo com o valor do input**

1. **Complexidade Logarítmica O(log n)**

**-** medida de desempenho de um algoritmo que acontece quando o comprimento da entrada é reduzido pela metade a cada passo.

1. **Complexidade Quadrática O(N²)**

**-** algoritmo que acontece quando o comprimento da entrada é reduzido pela metade a cada passo.

1. **Complexidade Cúbica: O(n3)O(n^3)O(n3)**

- algoritmo que de reduzi a entrada por um fator de 3 a cada passo.

1. **Complexidade Exponencial: O(2n) O(2^n)O(2n)**

- Algoritmo que o número de operações cresce exponencialmente em relação ao tamanho da entrada n.

1. **Complexidade Fatorial: O(n!) O(n!)O(n!)**

**-** Algoritmo que os números de operações crescem quando o fatorial do input.

1. **Complexidade Raiz Quadrada: O(n)O(\sqrt{n})O(n​)**

- algoritmos onde o número de operações é proporcional à raiz quadrada do tamanho da entrada do input

1. **Complexidade Polinomial: O(nk)O(n^k)O(nk)**

**- algoritmo tem funciona em torno de um polinômio de grau K**

1. **Complexidade Sublinear: O(log⁡n)O(\sqrt{\logn})O(logn​) ou similar**

**-** algoritmo que quando o número de operações de um algoritmo cresce mais devagar do que o tamanho da entrada do input

Merge shot:

O algoritmo Merge Sort é um algoritmo de divisão e conquista que classifica uma lista primeiro dividindo em sublistas, e depois reconstruindo-as e classificando em uma lista única

Complexidade de algoritmo:

**Complexidade Logarítmica O(log n)**

Insertion Sort :

 Algoritmo que usa uma parte de uma lista para armazenar os valores classificados e a outra parte da lista para armazenar os valores que ainda não foram classificados.

Complexidade de algoritmo:

**Constantes O(1)**

**Complexidade Linear O(n)**

**Complexidade Quadrática O(N²)**

Bubble Sort:

O **Bubble Sort** é um dos algoritmos de ordenação mais simples, mas também um dos menos eficientes para listas grandes. Ele funciona comparando elementos adjacentes da lista e trocando-os de posição se estiverem fora de ordem. O processo é repetido até que a lista esteja totalmente ordenada.

Complexidade de algoritmo:

**Complexidade Linear O(n)**

**Complexidade Quadrática O(N²)**

**Constantes O(1)**

Busca Linear

um algoritmo que procura um elemento em uma lista, percorrendo-a sequencialmente até encontrar o elemento do valor do input .

Complexidade de algoritmo:

**Complexidade Linear O(n)**

**Constantes O(1)**

Busca Binária

 um eficiente algoritmo para encontrar um item em uma lista ordenada de itens.

Complexidade de algoritmo:

**Constantes O(1)**

**Complexidade Logarítmica O(log n)**