

Estrutura de Dados - Lista (1): Complexidade

Prof. Rodrigo de Sapienza Luna

Turma 1

21 de agosto de 2025

Questões e Exercícios sobre Complexidade Computacional

Questões Teóricas

1. Defina o conceito de *complexidade de tempo* de um algoritmo.
2. O que significa dizer que um algoritmo tem complexidade de tempo $O(n^2)$?
3. Se um algoritmo tem complexidade de tempo $O(1)$, como seu tempo de execução varia em relação ao tamanho da entrada?
4. Qual é a diferença entre as notações $O(n)$ e $\Omega(n)$ na análise de algoritmos?
5. Dê um exemplo de um algoritmo que tenha complexidade de tempo $O(n \log n)$.
6. Se um algoritmo tem complexidade de espaço $O(n)$, o que isso nos diz sobre o uso de memória desse algoritmo?
7. Explique a importância de considerar a complexidade de tempo ao escolher um algoritmo para resolver um problema.
8. Qual é a complexidade de tempo de uma busca linear em uma lista desordenada? Justifique sua resposta.
9. O que significa dizer que a complexidade de tempo de um algoritmo é $O(n!)$? Dê um exemplo de problema com essa complexidade.
10. Se um algoritmo A tem complexidade de tempo $O(n)$ e outro algoritmo B tem complexidade de tempo $O(n^2)$, qual é a principal diferença em termos de eficiência?

Exercícios Práticos em C++

1. Considere o código C++ abaixo. Qual é a complexidade de tempo do algoritmo? Justifique sua resposta.

```
1 void exemplo1(int n) {  
2     for(int i = 0; i < n; i++) {  
3         std::cout << i << std::endl;  
4     }  
5 }
```

2. Analise o seguinte código. Qual é a complexidade de tempo e por quê?

```

1 void exemplo2(int n) {
2     for(int i = 0; i < n; i++) {
3         for(int j = 0; j < n; j++) {
4             std::cout << i * j << std::endl;
5         }
6     }
7 }

```

3. Determine a complexidade de tempo do código abaixo e explique.

```

1 void exemplo3(int n) {
2     for(int i = 1; i < n; i *= 2) {
3         std::cout << i << std::endl;
4     }
5 }

```

4. O código abaixo implementa uma busca binária. Qual é a complexidade de tempo? Explique sua análise.

```

1 int buscaBinaria(int arr[], int l, int r, int x) {
2     while (l <= r) {
3         int m = l + (r - l) / 2;
4
5         // Verifica se x est presente no meio
6         if (arr[m] == x)
7             return m;
8
9         // Se x for maior, ignora a metade esquerda
10        if (arr[m] < x)
11            l = m + 1;
12
13        // Se x for menor, ignora a metade direita
14        else
15            r = m - 1;
16    }
17
18    // Se chegar aqui, o elemento n o estava presente
19    return -1;
20 }

```

5. Para o código abaixo, qual é a complexidade de tempo? Justifique sua resposta.

```

1 void exemplo5(int n) {
2     for(int i = 0; i < n; i++) {
3         for(int j = 0; j < i; j++) {
4             std::cout << i + j << std::endl;
5         }
6     }
7 }

```