



**Lista 02**

**Prazo para entrega: 28-06-2023**

## 1 Instruções

- A entrega deve ser realizada presencialmente no início da aula.
- A lista deve ser feita à mão (caso você tenha um tablet com caneta stylus, você pode fazer usando o table e então entregar uma versão impressa da lista).
- A lista deve ser feita à caneta e sem resura.
- A lista é um trabalho individual.

## 2 Exercícios

1. Descreva, usando notação assintótica, um limitante superior para o tempo de execução do algoritmo a seguir. Justifique sua resposta.

---

### Algorithm 1 Abacate

---

```
1: Função ABACATE( $A, B, n$ )
2:    $k \leftarrow 0$ 
3:    $i \leftarrow 1$ 
4:   Enquanto  $i \leq n$  faça
5:      $k \leftarrow k + 1$ 
6:      $j \leftarrow i + 1$ 
7:     Enquanto  $j \leq n$  e  $A[j] < B[i]$  faça
8:        $j \leftarrow j + 1$ 
9:      $i \leftarrow j$ 
10:  Devolve  $k$ 
```

---

2. Forneça um limitante superior justo, usando notação assintótica, para os seguintes algoritmos. Justifique sua resposta.

```
int PowersOfTwo(int n) {
    int i;
    i = 0;
    while (n%2 == 0) {
        n = n/2;
        i++;
    }
    return i;
}
```

3. Suponha que  $T(1) = 1$  e  $T(2) = 1$ . Resolva as seguintes recorrências com notação  $O$  com o resultado mais justo possível (quando não especificado, use o método que preferir):

(a)  $T(n) = T\left(\frac{n}{3}\right) + n^2$  (método de iteração)

(b)  $T(n) = 4T\left(\frac{n}{2}\right) + n$  (método de substituição, suponha  $T(n) = O(n^2)$ )

(c)  $T(n) = 4T\left(\frac{n}{2}\right) + \sqrt{n}$  (árvore de recursão e método mestre)