

Atenção:

- Está sendo fornecido um arquivo Python com nome `paa2025_simulado.py` junto com esse enunciado.
- A solução da questão 1 deverá ser escrita no espaço destinado para ela no comentário da função `question_1` do arquivo Python.
- As soluções das questões 2, 3 e 4 deverão ser escritas na sua respectiva função do arquivo Python.
- O nome do arquivo Python e a assinatura das funções não devem ser alteradas.
- Caso seja necessário definir uma classe ou função auxiliar, escreva no escopo da função da questão que você estiver resolvendo.

Questão 1

Determine uma função $f(n)$ para o código a seguir tal que $T(n) = \theta(f(n))$. Explique como a solução foi encontrada.

```
HashTable t;

void f(int v[], int startInx, int endInx) {
    if (startInx < endInx - 1) {

        int midInx = (startInx + endInx) / 2;
        f(v, startInx, midInx);
        f(v, midInx, endInx);

        for (int i=startInx; i < endInx; i++) {
            t[v[i]]++;
        }

        int r[endInx - startInx];
        int aInx = startInx;
        int bInx = midInx;
        int rInx = 0;

        while (aInx < midInx && bInx < endInx) {
            r[rInx] = v[aInx] <= v[bInx] ? v[aInx++] : v[bInx++];
            r[rInx++] += t[r[rInx]];
        }

        while (aInx < midInx) {
            r[rInx] = v[aInx++];
            r[rInx++] += t[r[rInx]];
        }
    }
}
```

```
while (bInx < endInx) {  
    r[rInx] = v[bInx++];  
    r[rInx++] += t[r[rInx]];  
}  
  
for (aInx = startInx; aInx < endInx; ++aInx) {  
    v[aInx] = r[aInx - startInx];  
}  
}
```

Questão 2

Dada uma sequência v ordenada com n números distintos, projete um algoritmo que encontre os k elementos mais próximos do valor contido em uma posição x (os parâmetros são v , k e x). A proximidade entre os números nas posições i e j é calculada como $|a_i - a_j|$. O retorno deverá ser uma lista contendo os elementos mais próximo do elemento na posição x . O algoritmo deverá apresentar uma complexidade $O(k)$.

Questão 3

Um mergulhador precisa explorar x metros de um túnel subaquático. Seu tanque de oxigênio, quando cheio, permite mergulhar y metros. Existem pontos de reabastecimento de oxigênio ao longo do túnel. Dado um vetor ordenado com as distâncias dos pontos de reabastecimento desde a entrada do túnel, projete um algoritmo que determine em quais pontos o mergulhador deve reabastecer para completar a exploração com o menor número de paradas possível, ou indicar que a exploração é impossível. Analise a complexidade do algoritmo proposto.

Questão 4

Uma sequência A contém n inteiros positivos. Cada número em A pertence ao conjunto $\{n^2, n^2 + 1, n^2 + 2, \dots, n^2 + n\}$. Crie um algoritmo capaz de encontrar o número de A que mais se repete (caso exista mais de um pode retornar qualquer um dos números empatados). A complexidade do pior caso deverá ser $O(n)$.