

1) O algoritmo abaixo recebe o início de uma lista ligada e devolve VERDADEIRO se a lista ligada possui um ciclo e FALSO caso contrário.

```

Tem-Ciclo( p )
1. s ← p
2. t ← p
3. enquanto t ≠ NIL faça
4.     t ← prox( t )
5.     se t = s
6.         então devolve VERDADEIRO
7.     se t ≠ NIL
8.         então t ← prox(t)
9.     se t = s
10.        então devolve VERDADEIRO
11.    s ← prox( s )
12. devolve FALSO

```

a) Explique porque o algoritmo está correto.

b) Denote por **n** o número de elementos na lista apontada por **p**. Analise a complexidade do algoritmo, indicando seu comportamento assintótico. Justifique suas respostas.

2) Suponha que, para entradas de tamanho  $n$ , você tenha que escolher entre os algoritmos A e B.

- Algoritmo A resolve problemas dividindo-os em cinco problemas de metade do tamanho, recursivamente resolve cada subproblema e então combina as soluções em tempo  $O(n)$ .
- Algoritmo B resolve problemas dividindo-os em nove subproblemas de tamanho  $n/3$ , recursivamente resolve cada subproblema e então combina as soluções em tempo  $O(n^2)$ .

Estime o consumo de tempo de cada um desses algoritmos? Qual algoritmo é assintoticamente mais eficiente no pior caso? Justifique suas respostas.

3) Seja  $A[1..n]$  um vetor de inteiros. Um segmento  $A[i..k]$  ( $1 \leq i \leq k \leq n$ ) é não decrescente se  $A[i] \leq A[i+1] \leq \dots \leq A[k-1] \leq A[k]$ . O comprimento de um tal segmento é  $k-i+1$ . Escreva um algoritmo que receba um vetor  $A[1..n]$  e devolve o comprimento do segmento não decrescente de comprimento máximo de  $A$ . Por exemplo, se  $A = (1, 4, 8, 9, 2, 3, 4, 5, 6, 8, 9, 1, 2, 3)$  então o segmento não decrescente máximo é  $M = (2, 3, 4, 5, 6, 8, 9)$  com comprimento 7.

4) Escreva um algoritmo que recebe uma matriz  $M_{n \times n}$  de inteiros e encontra um caminho de  $M_{1,n}$  até  $M_{n,1}$  de forma a minimizar a soma dos valores absolutos das diferenças entre elementos consecutivos do caminho. Um caminho é definido por movimentos horizontais ou verticais entre elementos adjacentes da matriz, por exemplo, para a matriz abaixo, o custo mínimo é 66.

<u>1</u>	<u>17</u>	<u>6</u>	46	<u>7</u>
<u>4</u>	58	<u>15</u>	<u>6</u>	<u>10</u>
<u>2</u>	8	61	18	29
<u>9</u>	18	6	23	9
<u>11</u>	6	13	41	12

5) Projete um algoritmo para construir um heap que contém todos os elementos de dois heaps de tamanho  $n$  e  $m$ , respectivamente. Os heaps são dados em listas ligadas. No pior caso seu algoritmo deverá rodar em ordem  $O(\lg(m+n))$ .