

# Exercícios: Pilhas, Filas, Listas Encadeadas e Ordenação - Unidade 01 - Prova 1

## Algumas considerações

As questões abaixo servirão de base para a sua prova. Não significa que as questões da prova serão iguais aquelas listadas abaixo, e, também, não significa que serão diferentes.

### Pilhas e Filas com Vetores ou com Ponteiros

1. Descreva o algoritmo(ou código) que insira um elemento em uma pilha com vetores.
2. Descreva o algoritmo(ou código) que remova um elemento em uma pilha com vetores.
3. Descreva o algoritmo(ou código) que insira um elemento em uma pilha com vetores que permita redimensionamento.
4. Descreva o algoritmo(ou código) que remova um elemento em uma pilha com vetores que permita redimensionamento.
5. Descreva o algoritmo(ou código) que insira um elemento em uma pilha com ponteiros (lista encadeada).
6. Descreva o algoritmo(ou código) que remova um elemento em uma pilha com ponteiros (lista encadeada).
7. Descreva o algoritmo(ou código) que insira um elemento em uma fila com vetores.
8. Descreva o algoritmo(ou código) que remova um elemento em uma fila com vetores.
9. Descreva o algoritmo(ou código) que insira um elemento em uma fila com vetores que permita redimensionamento.
10. Descreva o algoritmo(ou código) que remova um elemento em uma fila com vetores que permita redimensionamento.
11. Descreva o algoritmo(ou código) que insira um elemento em uma fila com ponteiros (lista simplesmente encadeada).
12. Descreva o algoritmo(ou código) que remova um elemento em uma fila com ponteiros (lista simplesmente encadeada).
13. Descreva a saída resultante da seguinte série de operações sobre uma pilha: push(3), pop(), push(2), push(8), pop(), pop(), push(9), push(1), pop(), push(7), push(6), pop(), pop(), push(4), pop(), pop().
14. Descreva a saída resultante da seguinte série de operações sobre uma fila: enqueue(5), enqueue(3), dequeue(), enqueue(2), enqueue(8), dequeue(), dequeue(), enqueue(9), enqueue(1), dequeue(), enqueue(7), enqueue(6), dequeue(), dequeue().

## Listas Encadeadas - Unidade 01

15. Quais são os passos necessário para inserir um novo item na cabeça de uma lista encadeada? Use uma frase em português para cada passo.
16. Suponha que  $p$  é uma referência a um objeto nó em uma lista encadeada com inteiros. Quais são os passos para remover o nó seguinte ao nó  $p$ ? Use uma frase em português para cada passo.
17. Escreva um procedimento que retorna o próximo nó de uma lista encadeada sem sentinela caso esse nó existe ou retorna nulo caso contrário.
18. Implemente o método *count33s(cabeça)* para uma lista encadeada de inteiros, o método assume que *cabeça* é o endereço da cabeça de uma lista duplamente encadeada, a lista pode estar vazia ou não. O procedimento deve retornar o número de vezes que o número 33 aparece na lista.
19. Implemente o método *all33s(cabeça)* para uma lista encadeada de inteiros, o método assume que *cabeça* é o endereço da cabeça de uma lista duplamente encadeada, a lista pode estar vazia ou não. O procedimento deve retornar *true* se todos os dados na lista são 33 e falso caso contrário.
20. Escreva um algoritmo recursivo para imprimir do começo para o fim em uma lista encadeada não circular.
21. Escreva um algoritmo recursivo para imprimir do fim para o começo em uma lista encadeada não circular.
22. Descreva um método não recursivo para encontrar o nó do meio de uma lista duplamente encadeada com sentinelas.
23. Suponha que esteja lidando com uma lista encadeada de inteiros. Escreva uma função contagem. Ela deve receber a lista e um inteiro  $a$ . Ao final retornará o número de vezes que o valor de  $a$  ocorre na lista.
24. Escreva uma função *getNth* para uma lista encadeada. A função recebe um inteiro, *index*, que funciona como um índice, a função retorna o elemento na posição *index*. Por exemplo suponha uma lista encadeada com a seguinte sequência de elementos  $L = \{12, 17, 36, 100, 666\}$ , se a função for executado sobre a lista e recebendo *index* = 2, então ela deve retornar 17.
25. Escreva uma função *InsiraApos*. Ela recebe um inteiro, *cnt*, e um novo elemento,  $x$ , do tipo nó, a ser inserido na lista. Este elemento deve ser inserido após o número armazenado em *cnt*. Por exemplo, se *cnt* = 2 deve-se inserir  $x$  após o segundo elemento da lista. Se *cnt* = 0 então  $x$  será o novo primeiro elemento da lista.

26. Escreva uma função anexar. Anexar recebe duas listas duplamente encadeadas sem sentinelas do mesmo tipo,  $A$  e  $B$ . O procedimento anexar concatena as duas listas. Ou seja, a lista  $A$  se torna a junção das duas listas,  $A$  e  $B$ .
27. Suponha que um nó  $x$  de uma lista duplamente encadeada não ordenada tenha sido removido. Nenhum campo de  $x$  foi alterado e nenhuma outra operação foi realizada sobre a lista. Escreva um procedimento que o re-insere exatamente na mesma posição que ele estava na lista. (Dica: leia sobre o algoritmo dancing links de Knuth).

## Ordenação - Unidade 01

28. Qual algoritmo de ordenação é o mais rápido quando o array de entrada já está ordenado?
29. Escreva os pseudo-códigos de duas ordenações que não utilizem comparações entre os elementos do vetor.
30. Suponha que um array de inteiros  $A$  está sendo ordenado usando quicksort. Após a primeira chamada de partição tem-se  $A = \{2, 4, 1, 5, 9, 12, 15, 10\}$ . Com base nessas informações, qual das afirmações abaixo pode ser afirmada sem dúvidas?
- A. O número 5 foi o elemento pivô usado.
  - B. O elemento pivô foi ou o 5, ou o 9.
  - C. O número 9 serviu como pivô.
  - D. Nem o 5, nem o 9 serviram como pivô.
31. Escreva os pseudo-códigos de dois algoritmos de ordenação que utilize comparação, um recursivo e outro não recursivo. Inclua comentários descrevendo os algoritmos.
32. Escreva o pseudo-código do quicksort determinístico cujo pivô será sempre o primeiro elemento da partição.
33. Descreva o pior caso e o melhor caso de cada algoritmo abaixo:
- Ordenação por inserção
  - Mergesort
  - Quicksort
  - Ordenação por seleção
34. Escreva uma adaptação de cada um dos algoritmos listados abaixo para ordenarem do maior elemento para o menor.
- Ordenação por inserção
  - Mergesort
  - Quicksort

- Ordenação por seleção
35. Escreva uma adaptação de cada um dos algoritmos listados abaixo para ordenarem segundo as seguintes regras: dados dois números, se os dois forem pares então o maior dos dois deve vir primeiro, se os dois forem ímpares então o menor deles deve aparecer antes, se um deles for par e o outro for ímpar então deve-se arranjá-los em ordem crescente. Os algoritmos a adaptar são os seguintes:
- Ordenação por inserção
  - Mergesort
  - Quicksort
  - Ordenação por seleção
36. Dado o vetor  $A = \langle 1, 4, 5, 8, 2, 6, 9, 7 \rangle$  e os valores  $p = 1$ ,  $q = 4$  e  $r = 8$ . O Algoritmo Merge conseguirá ordenar o vetor  $A$  se receber  $p$ ,  $q$  e  $r$  como especificados? Sim? Não? Por que?

## Outras - Unidade 01 - Prova 1 e 2

37. No desenvolvimento de um software que analisa bases de DNA, representadas pelas letras A, C, G, T, utilizou-se as estruturas de dados: pilha e fila. Considere que, se uma sequência representa uma pilha, o topo é o elemento mais à esquerda; e se uma sequência representa uma fila, a sua frente é o elemento mais à esquerda.
- Análise o seguinte cenário: a sequência inicial ficou armazenada na primeira estrutura de dados na seguinte ordem:  $(A, G, T, C, A, G, T, T)$ . Cada elemento foi retirado da primeira estrutura de dados e inserido na segunda estrutura de dados, e a sequência ficou armazenada na seguinte ordem:  $(T, T, G, A, C, T, G, A)$ . Finalmente, cada elemento foi retirado da segunda estrutura de dados e inserido na terceira estrutura de dados e a sequência ficou armazenada na seguinte ordem:  $(T, T, G, A, C, T, G, A)$ .
- Qual a única sequência de estruturas de dados apresentadas a seguir pode ter sido usada no cenário descrito acima?
- A. Fila - Pilha - Fila.
  - B. Fila - Fila - Pilha.
  - C. Fila - Pilha - Pilha.
  - D. Pilha - Fila - Pilha.
  - E. Pilha - Pilha - Pilha.
38. Suponha uma pilha implementada usando arrays (vetores) com no máximo 6 elementos e inicialmente vazia. Descreva ou desenhe o estado dos elementos armazenados em tal

pilha, bem como o valor do topo, após a seguinte sequência de operações  $\text{push}(S,4)^1$ ,  $\text{push}(S,1)$ ,  $\text{pop}(S)^2$ ,  $\text{push}(S,8)$  e  $\text{pop}(S)$ . E se fosse uma fila, como ficaria?

39. Descreva o princípio dividir e conquistar.
40. Descreva as fases de dividir, conquistar e combinar dos algoritmos quicksort e mergesort.

---

<sup>1</sup>push = empilhar  
<sup>2</sup>pop = desempilhar