

Parte teórica. Duração: 30m

Nome: _____ Código: _____

Notas: - Responda às questões seguintes, indicando a opção correta (em maiúsculas)
- Cada resposta errada vale -20% da cotação da pergunta

1. Considere um vetor de n inteiros positivos a_1, a_2, \dots, a_n . Pretende-se determinar se existe uma sequência de elementos deste vetor cuja soma seja igual a s . Para a resolução deste problema, é criado um vetor auxiliar de tamanho n , em que a posição i é igual a $a_1 + a_2 + \dots + a_i$. Qual o modo mais eficiente de pesquisar s neste vetor?
- A. Pesquisa linear
 - B. Pesquisa binária
 - C. Pesquisa aleatória
 - D. Pesquisa bidirecional (duas pesquisas simultâneas: uma a partir do início do vetor, outra a partir do final)
 - E. É indiferente

Resposta: B

2. No excerto (pseudo-código) apresentado a seguir, inicialmente $it1$ e $it2$ são referências para um mesmo nó de uma lista ligada circular. Em que condições é que o ciclo termina?

```
do {  
    it1 = elemento_apos(it1);  
    it2 = elemento_apos(elemento_apos(it2));  
} while (it1 != it2)
```

- A. Sempre
- B. Apenas se $it1 = \text{elemento_apos}(it2)$
- C. Apenas se a lista tiver um número ímpar de elementos
- D. Apenas se a lista contém dois elementos iguais
- E. Nenhuma das possibilidades anteriores

Resposta: A

3. Qual a estrutura de dados linear mais adequada (mais eficiente em tempo e em uso de espaço) a usar quando se pretende procurar, num conjunto de elementos, determinado elemento X , ou, no caso deste não existir, o imediatamente anterior.

- A. Lista ordenada simplesmente ligada
- B. Lista ordenada duplamente ligada
- C. Lista desordenada simplesmente ligada
- D. Lista desordenada duplamente ligada
- E. Indiferente

Resposta: A

4. Qual a estrutura de dados mais adequada a usar na gestão de pacientes de um laboratório de análises, que vão efetuar recolha de sangue em determinado dia?

- A. Pilha
- B. Fila
- C. Lista
- D. Lista ordenada
- E. Indiferente

Resposta: B

Parte teórica. Duração: 30m

5. A estrutura fila é implementada através de uma lista ligada, que mantém referências para o primeiro nó (*inicio*) e último nó (*fim*) da lista. Qual destas referências pode ser alterada aquando da inserção de um elemento na fila?
- A. Ambas as referências, se a fila estiver vazia
 - B. Apenas a referência *inicio*, para qualquer fila (vazia ou não)
 - C. Apenas a referência *fim*, para qualquer fila (vazia ou não)
 - D. Nenhuma das referências, se a fila estiver vazia
 - E. Nenhuma das possibilidades anteriores

Resposta: A

6. Qualquer estrutura de dados linear:
- A. Mantém a ordem temporal (ordem de inserção/remoção) dos seus elementos
 - B. Possui tempo de acesso linear aos seus elementos
 - C. Possui tempo de inserção linear
 - D. Permite o uso de iteradores
 - E. Nenhuma das possibilidades anteriores

Resposta: E

7. Qual a função principal de um iterador numa coleção de elementos?
- A. Permitir a inserção de elementos na coleção
 - B. Permitir percorrer os elementos da coleção de forma sequencial
 - C. Permitir a remoção de elementos na coleção
 - D. Permitir alterar um qualquer elemento da coleção
 - E. Nenhuma das possibilidades anteriores

Resposta: B

8. É verdade que posso implementar uma pilha (*stack*) e as suas operações de *pop()*, *top()* e *push()*:
- A. Usando um `vector` do C++ (chamado `vec`) em que: `vec.push_back()` implementa o `push()`; `vec[vec.size()-1]` implementa o `top()`; `vec.erase(vec.size()-1)` implementa o `pop()`
 - B. Usando uma fila em que: a operação de inserir na fila implementa o `push()` da pilha; retirar da fila implementa o `pop()` da pilha; retirar da fila implementa o `top()` da pilha
 - C. Usando uma fila em que: a operação de inserir na fila implementa o `push()` da pilha; retirar da fila implementa o `pop()` da pilha; usar o `top()` da fila implementar o `top()` da pilha.
 - D. Usando uma lista ligada em que: inserir no início da lista implementa o `push()` da pilha; retirar o elemento do início da lista implementa o `pop()` da pilha; devolver o elemento da cabeça da lista (sem eliminar) implementa o `top()` da pilha
 - E. Nenhuma das possibilidades anteriores

Resposta: D

Parte teórica. Duração: 30m

Nome: _____ Código: _____

9. Tendo um vetor completo (com todas as posições preenchidas) e com os elementos ordenados por ordem crescente, para guardar os mesmos elementos no mesmo vetor mas ordenados por ordem decrescente devo:
- A. Percorrer o vetor começando na posição zero e fazendo `push()` dos seus elementos numa pilha. Fazer `pop()` da pilha até a esvaziar, colocando os elementos no vetor começando na última posição do vetor e terminando na posição zero.
 - B. Percorrer o vetor começando na posição zero e fazendo `push()` dos seus elementos numa pilha. Fazer `pop()` da pilha até a esvaziar, colocando os elementos no vetor começando na posição zero do vetor e terminando na última posição.
 - C. Retirar todos os elementos do vetor começando na posição zero e fazendo `push()` numa fila. Fazer `pop()` da fila, fazendo `push_back()` no vetor.
 - D. Retirar todos os elementos do vetor começando na última posição e fazendo `push()` numa pilha. Fazer `pop()` da pilha, fazendo `push_back()` no vetor.
 - E. Nenhuma das possibilidades anteriores

Resposta: B

10. Qual a afirmação correta?

- A. Filas e pilhas são estruturas iguais porque ambas têm (na implementação STL) as operações `push()` e `pop()`
- B. Para a mesma sequência de dados, fazendo a mesma sequência de operações `push()` e `pop()` numa pilha e numa fila, obtém-se a mesma sequência de dados de saída
- C. Para a mesma sequência de dados, as sequências de operações I) e II) resultam na mesma sequência de dados de saída
 - I) inserir a sequência numa pilha; retirar todos os elementos da pilha
 - II) inverter a sequência inicial de dados; inserir a nova sequência numa fila; retirar todos os elementos da fila
- D. Numa pilha, as operações `top()` e `pop()` produzem o mesmo efeito no conteúdo da pilha
- E. Nenhuma das possibilidades anteriores

Resposta: C