

Exame: Quiz #06

Resultado obtido não conta para a sua avaliação.  
Apenas uma das respostas está correcta por pergunta.  
Podem ser escolhidas várias respostas a uma mesma pergunta.

Aluno: Rodrigo João Bastos Mendes

Pode tornar a responder quantas vezes quiser ao questionário.  
O tempo mostrado é só para uma tentativa de resposta a todo o questionário.  
Cada questionário estará online para efeitos de frequência durante um curto período de tempo.

Para efeitos informativos, é mostrada uma pontuação:  
1 resposta correta: 1 pontos  
1 resposta errada: -0.5 pontos

Valorização: 8.5  
Correcção: as afirmações correctas estão marcadas com C e as erradas estão marcadas com E

Sair

## 1. Listas Ligadas Circulares I

Numa lista ligada circular, o **último nó** tem o atributo *next* a apontar para:

☐ E ~~sempre para si próprio~~

☐ E ~~sempre para *null*~~

☒ C **sempre para o primeiro nó**

## 2. Listas Ligadas Circulares II

Numa lista ligada circular implementada com atributos *last* e *size* (como feito na aula teórica), **qual das seguintes operações é menos eficiente** e depende do tamanho da lista?

☐ E ~~"Rodar" a lista (fazer avançar para o início o segundo elemento, passando o anterior primeiro para último)~~

☐ E ~~Remover o primeiro elemento~~

☒ C **Remover o último elemento**

## 3. Listas Duplamente Ligadas I

Qual a diferença de uma **lista duplamente ligada** para uma lista ligada simples?

☐ E ~~Cada nó "aponta" para dois nós seguintes~~

☒ C **Cada nó "aponta" para o nó seguinte e também para o nó anterior**

☐ E ~~Cada nó "aponta" duas vezes para o mesmo nó seguinte~~

## 4. Listas Duplamente Ligadas II

No contexto de listas duplamente ligadas, o que são **nós sentinelas**?

☐ E ~~Nós que contêm valores que queremos vigiar, sendo desse modo importantes para a topologia da lista~~

☒ C **Nós "fictícios" adicionados no início e no fim da lista que permitem simplificar a implementação, deixando de haver casos excepcionais por exemplo para remoções e inserções**

☐ E ~~Nós que para além de apontarem para o próximo nó, também apontam para o nó anterior~~

## 5. Listas Duplamente Ligadas III

Suponha que tem uma lista duplamente ligada onde algures no meio da lista aparecem consecutivamente os nós *n1*, *n2* e *n3*. Como podemos **remover o nó n2** da lista?

☐ E ~~Basta colocar n2.prev e n2.next a apontar para *null*~~

☒ C **Basta colocar n1.next a apontar para n3 e n3.prev a apontar para n1**

☐ E ~~Basta colocar n1.next a apontar para n2 e n2.prev a apontar para n1~~

## 6. Pilhas e Filas I

Qual dos seguintes tipos abstractos de dados é **LIFO** (*Last In, First Out*)

☒ C **Pilha (*Stack*)**

☐ E ~~Fila (*Queue*)~~

☐ E ~~Fila com dois extremos (*Deque*)~~

## 7. Pilhas e Filas II

Suponha que ['A','B','C'] representa uma pilha com A no topo da pilha. Qual dos seguintes **conjuntos de operações** produziria uma pilha s com esse conteúdo?

☐ E ~~s.pop("ABC")~~

☐ C **s.push('C'); s.push('B'); s.push('A');**

☒ E ~~s.push('A'); s.push('B'); s.push('C');~~

## 8. Pilhas e Filas III

Suponha que inseriu consecutivamente os números 1, 2, 3 e 4 numa pilha. Se retirar agora os elements um a um (fazendo *pop()*'s), **em que ordem é que eles saem da pilha?**

☐ E ~~1, 4, 2, 3~~

☒ C **4, 3, 2, 1**

☐ E ~~1, 2, 3, 4~~

## 9. Pilhas e Filas IV

Numa fila, a operação *dequeue* serve para:

☐ E ~~transformar a fila num *deque*~~

☒ C **retirar um elemento no extremo oposto onde os elementos são inseridos**

☐ E ~~retirar um elemento no mesmo extremo onde os elementos foram inseridos~~

## 10. Pilhas e Filas V

Imagine um professor em horário de atendimento, esclarecendo dúvidas de alunos que vão chegando, sendo que atende um por um, começando por quem chegou primeiro, depois por quem chegou em segundo lugar, e por aí adiante. **Qual dos seguintes tipos abstractos de dados seria o mais natural para corresponder a este processo?**

☐ E ~~Pilha (*Stack*)~~

☐ E ~~Fila com dois extremos (*Deque*)~~

☒ C **Fila (*Queue*)**