

Listas

Total de pontos 90/100

O endereço de e-mail do participante (**douglasam@rede.ulbra.br**) foi registrado durante o envio deste formulário.

Pontuação da seção 90/100

✓ 1. Uma Lista Estática Sequencial tem como desvantagens (pode haver mais de uma alternativa correta): *

 10/10

- ☐ Acesso indexado aos dados
- ☒ Movimentação dos dados, especialmente nas inserções/remoções feitas no início da lista ✓
- ☐ Inserção/remoção de elemento ao fim da lista
- ☒ Não ser possível aumentar o tamanho da lista em tempo de execução ✓
- ☐ Possibilidade de modificar a quantidade de elementos da lista a qualquer momento



✓ 2. Foi informado a você que a função apresentada na imagem esvazia uma Lista Estática Sequencial. Você ... *

10/10

```
def esvaziarLista(self):  
    self.quant = 0
```

- ☐ Discorda, pois é necessário também preencher toda a lista com None para esvaziar a Lista
- ☐ Discorda, pois os atributos prim e ult devem receber None também
- ☐ Concorde em parte, pois esta função só funcionará se a lista já estiver cheia
- ☐ Concorde em parte, pois esta função só funcionará se só houver um elemento na lista
- ☒ Concorde totalmente, pois o atributo quant gerencia também a inserção e remoção de elementos da lista

✓

✓ 3. Considerando uma Lista Dinâmica Simplesmente Encadeada, a inserção no início e no fim da lista são igualmente "baratas" considerando a complexidade de sua implementação *

10/10

- ☒ Sim
- ☐ Não

✓

✓ 4. Considerando uma Lista Dinâmica Simplesmente Encadeada, a remoção no início e no fim da lista são igualmente "baratas" considerando a complexidade de sua implementação *

10/10

- ☐ Sim
- ☒ Não

✓



- ✓ 5. (VUNESP – 2013 - MPE-ES - Agente Especializado - Analista de Sistemas) Considere a estrutura de dados do tipo Lista Encadeada, em que os elementos são alocados na memória à medida que são inseridos na lista, e removidos à medida que são dela retirados. Em um dado momento, essa lista possui 5 elementos (E1, E2, E3, E4 e E5), de forma que E1 aponta para E2, E2 aponta para E3, E3 aponta para E4 e E4 aponta para E5. Deseja-se retirar o elemento E3 dessa lista. Com base nas informações apresentadas, assinale a alternativa que apresenta as ações necessárias para realizar essa operação. 10/10

- ☒ E2 passa a apontar para E4; Libera-se E3 ✓
- ☐ E2 passa a apontar para E3; Libera-se E2
- ☐ E3 passa a apontar para E2; Libera-se E3
- ☐ E4 passa a apontar para E2; Libera-se E4
- ☐ E4 passa a apontar para E3; Libera-se E4

- ✓ 6. O código abaixo, relacionado a uma Lista Estática Sequencial, pode ser chamado de: 10/10

```
def oque(self, valor):  
    for i in range(self.quant, 0, -1):  
        self.vetor[i] = self.vetor[i-1]  
    self.vetor[0] = valor  
    self.quant += 1
```

- ☒ inserirInicio ✓
- ☐ inserirFim
- ☐ removerInicio
- ☐ removerFim
- ☐ show

✓ 7. (CS-UFG - 2014 - UEAP - Analista de Tecnologia da Informação - Desenvolvimento de Sistemas) Não existe, a priori, uma maneira considerada a melhor para implementar listas lineares, quer seja por contiguidade física (sobre arranjos), quer seja por encadeamento. Entretanto, há vantagens na implementação de listas lineares por contiguidade física, como

10/10

- ☐ a movimentação de dados em operações de inserção e remoção no arranjo
- ☐ a necessidade de uma estimativa prévia do tamanho definido para o arranjo
- ☒ o acesso direto a qualquer elemento da lista por meio do índice no arranjo ✓
- ☐ o tempo de acesso a dados diretamente proporcional ao tamanho do arranjo
- ☐ nenhuma das alternativas anteriores, pois não há vantagens na implementação de listas lineares por contiguidade física

✗ 8. (CETAP - 2015 - MPCM – Técnico em Informática - Desenvolvimento) A estrutura de dados “Nó”, utilizada para definir listas ligadas, deve conter pelo menos:

0/10

- ☐ Um valor e uma ligação ao próximo nó
- ☐ Um valor e um método que defina se um valor é maior ao outro
- ☐ Um índice e um valor
- ☐ Um valor HASH e um vetor de valores
- ☒ Uma chave e um valor ✗

Resposta correta

- ☒ Um valor e uma ligação ao próximo nó



✓ 9. (FCC - 2009 - TJ-PI - Analista Judiciário - Análise de Sistemas – Desenvolvimento) Uma lista dinâmica é uma estrutura que corresponde a uma sequência lógica de entradas ou nós. Cada nó armazena a localização do próximo elemento na sequência, ou seja, de seu nó sucessor. Nessa estrutura,

- ☐ para estabelecer a ligação entre um nó já pertencente a uma lista e um novo nó, basta fazer com que o novo nó referencie, no camponext, o nó que anteriormente era referenciado pelo nó original, desde que esse campo não tenha o valor nulo.
- ☐ a existência de um ponteiro apontando para o 1º elemento e outro para o fim da lista permite que a inserção ou deleção de dados de um nó que esteja no meio da lista seja rapidamente executada.
- ☐ enquanto a entrada que determina o início da lista é mantida em uma estrutura descritora dessa lista, a entrada que marca o fim da lista é mantida fora dessa estrutura.
- ☐ o armazenamento de uma lista requer uma área contígua de memória para permitir a otimização no processamento de criação e remoção de nós da lista.
- ☒ o armazenamento de uma lista dinâmica não requer uma área contígua de memória. Normalmente são definidos procedimentos que permitem criar e remover nós na memória. ✓



✓ 10. O código abaixo, relacionado a uma Lista Dinâmica Simplesmente Encadeada, pode ser chamado de:

10/10

```
def fazoque(self, valor):  
    if self.quant==0:  
        self.prim = self.ult = No(valor, None)  
    else:  
        self.prim = No(valor, self.prim)  
    self.quant+=1
```

- ☐ inserirFim
- ☒ inserirInicio
- ☐ removerFim
- ☐ removerInicio
- ☐ esvaziaLista



Este formulário foi criado em Universidade Luterana do Brasil.

Google Formulários

