|  |  |  |  |
| --- | --- | --- | --- |
|  | | Avaliação Teórica  G1Timbre Ulbra1 G2 Timbre Ulbra1  Valor 2,0 | |
| Curso: Sistemas de Informação / Ciência da Computação / Engenharia de Software | Disciplina: Estruturas de Dados I | | Data: 22/09/2017 |
| Turma: 0305 | Professor: Fabiano Fagundes | | Nota: |
| Aluno: | | |

**Instruções:**

1. Leia atentamente as questões antes de respondê-las;
2. A interpretação é parte essencial da avaliação;

**Cada questão abaixo vale 0,4 pontos. Só há uma opção correta em cada uma. Devem ser respondidas sem rasuras, a caneta.**

**1.** Considerando a lista abaixo, quais seriam as impressões dos textos de códigos a seguir?

Obs: o símbolo ∄ (não existe) denota a impossibilidade de gerar uma impressão devido a algum erro, por exemplo.



1) print (self.prim.prox.prox.ant.prox.info);

self.prim=A

self.prim.prox=B

self.prim.prox.prox=C

self.prim.prox.prox.ant=B

**self.prim.prox.prox.ant.prox.info=C**

2) print(self.prim.ant.prox.ant.prox.info);

self.prim=A

self.prim.ant=E

self.prim.ant.prox=A

self.prim.ant.prox.ant=E

**self.prim.ant.prox.ant.prox.info=A**

3) print(self.ult.prox.prox.ant.prox.info);

self.ult=E

self.ult.prox=A

self.ult.prox.prox=B

self.ult.prox.prox.ant=A

**self.ult.prox.prox.ant.prox.info=B**

4) print(self.ult.prim.prox.ant.prox.info);

self.ult=E

**self.ult.prim=ERRO**

5) print(self.prim.ant.ant.ant.prox.info);

self.prim=A

self.prim.ant = E

Self.prim.ant.ant = D

self.prim.ant.ant.ant=C

**Self.prim.ant.ant.prox.info = D**

6) print(ult.prox.prox.info.prox.info);

ult.prox=A

ult.prox.prox.info=B

**ult.prox.prox.info.prox=ERRO**

Assinale a alternativa que apresenta as impressões corretas:

( ) a) 1 – B; 2 – B; 3 – A; 4 – B; 5 – E; 6 – C

( ) b) 1 – C; 2 – A; 3 – B; 4 – A e B; 5 – E; 6 – B e C

**( ) c) 1 – C; 2 – A; 3 – B; 4 – ∄; 5 – D; 6 – ∄**

( ) d) 1 – C; 2 – A; 3 – B; 4 – ∄; 5 – ∄; 6 – ∄

( ) e) Nenhuma das Alternativas

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

1. **Um valor e uma ligação ao próximo no.**

**“Nó” precisa de valor= informação, e uma ligação que irá guardar o endereço do nó seguinte.**

1. Um valor e um método que defina se um valor é maior ao outro.
2. Um indice e um valor.
3. Um valor HASH e um vetor de valores.
4. Uma chave e um valor

3. **(UPENET – 2012 – JUCEPE – Analista de Tecnologia da Informação)** Uma lista é uma estrutura de dados, que implementa uma coleção de valores. Analise as afirmativas sobre a implementação de listas.

I. A implementação de listas usando vetores utiliza um espaço de armazenamento contíguo, e a inserção de um elemento quando não feita ao final da lista requer movimentação.

II. Na implementação de listas com encadeamento simples, cada elemento possui apenas informação de quem é seu predecessor.

**Lista encadeamento simples possui apenas a informação de quem é sucessor ou seja o próximo.**

III. Na implementação de listas com encadeamento duplo, cada elemento possui a informação de quem é seu sucessor e predecessor.

Assinale a alternativa CORRETA

1. Somente I é verdadeira.
2. Somente II é verdadeira.
3. **Somente I e III são verdadeiras.**
4. Somente II e III são verdadeiras.
5. I, II e III são verdadeiras.

4. Responda verdadeiro ou falso para os itens a seguir:

( FALSO ) **(CESPE - 2008 - TRT - 5ª Região (BA) - Técnico Judiciário - Tecnologia da Informação)** A principal característica de uma lista simplesmente encadeada é o fato de o último elemento da lista apontar para o elemento imediatamente anterior.

**Na lista simplesmente encadeada o último elemento aponta para NULL;**

( VERDADE ) **(CESPE - 2009 - ANAC - Técnico Administrativo – Informática)** Em uma lista circular duplamente encadeada, cada nó aponta para dois outros nós da lista, um anterior e um posterior.

( FALSO ) **(CESPE - 2010 - Banco da Amazônia - Técnico Científico - Tecnologia da Informação - Análise de Sistemas)** Em uma lista encadeada, o tempo de acesso a qualquer um de seus elementos é constante e independente do tamanho da estrutura de dados.

**Quanto mais no final estiver o elemento procurado, mais tempo será gasto para percorrer a lista do início até esse elemento.**

( FALSO ) **(CESPE - 2012 – Banco da Amazônia – Técnico Científico - Administração de Dados)** As listas duplamente encadeadas diferenciam-se das listas simplesmente encadeadas pelo fato de, na primeira, os nós da lista formarem um anel com o último elemento ligado ao primeiro da lista.

**O conceito define uma lista circular e não uma duplamente encadeada**.

A opção que corresponde à ordem correta, respectivamente, das respostas dos itens acima é:

( ) a) FALSO, FALSO, VERDADE, VERDADE

**( ) b) FALSO, VERDADE, FALSO, FALSO**

( ) c) VERDADE, FALSO, VERDADE, FALSO

( ) d) FALSO, VERDADE, VERDADE, VERDADE

( ) e) FALSO, VERDADE, FALSO, VERDADE

5. **(CESGRANRIO – EPE – Técnico de Nível Superior - Área Tecnologia da Informação – 2006)**

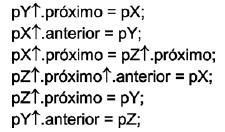
Os registros em uma lista, duplamente encadeada com 20 elementos possuem cada um três campos:

**próximo**: um ponteiro para o próximo elemento da lista;

**valor**: informação armazenada pelo elemento;

**anterior**: um ponteiro para o elemento anterior da lista.

Sendo "Z" o décimo elemento desta lista e "X" e "Y" dois outros elementos que não pertencem à lista, com seus respectivos ponteiros "pZ", "pX" e "pY", considere o trecho de código abaixo.



Este trecho de código é usado para inserir na lista os elementos:

1. **Y, logo após o Z, e X, logo após o Y.**
2. Y, antes do Z, e X, logo após o Z.
3. Y, antes do Z, e X, antes do Y.
4. X, logo após o Z, e Y, logo após o X.
5. X, antes do Z, e Y, logo após o Z.

A lista possui 20 elementos sendo o “Z” o décimo e “X” e “Y” elementos não pertencentes à lista:

Na primeira instrução y.prox() aponta para “X”, e na segunda o anterior de “X” aponta para “Y”.

3° - X.prox() aponta para z.prox() o 11° elemento da lista, irei representa-lo como “W”.

4° - W.ant() aponta para “X”.

5° - Z.prox() aponta para “Y”.

6° - Y.ant() aponta para “Z”.

Lista:

[...] Z<-> Y <-> X <-> W [...]

Logo:

**Y, logo após o Z, e X, logo após o Y.**