



# Teste de Conhecimento

avalie sua aprendizagem



## ESTRUTURA DE DADOS

8ª aula



Exercício: CCT0826\_EX\_AB\_202101110137\_V1

20/04/2022

Aluno(a): DOUGLAS MATOS DA SILVA

2022.1 EAD

Disciplina: CCT0826 - ESTRUTURA DE DADOS

202101110137

### 1 Questão

q Considere

```
int *p, *q, *r;  
int x = 10, y = 20;
```

Após seguir cada instrução na ordem dada a seguir:

```
p = &x;  
q = &y;  
(*q) = *p + *q;  
r = q;  
q = *p;
```

Respondido em 20/04/2022 08:31:49

#### Explicação:

Fazendo passo a passo temos:

```
int *p, *q, *r;  
int x = 10, y = 20;
```

```
p = &x; // p aponta para x, que vale 10
```

```
q = &y; // q aponta para y, que recebeu 20
```

```
*p = *p + *q; // a área apontada por p, que é x, vai receber 10 + 20, que dá 30. Logo, *p e x valem 30.
```

```
(*q)++; // a área apontada por q é incrementada. Ou seja, y ou *q recebem 21
```

```
r = q; // r aponta para y
```

```
q = p; // q aponta para onde p aponta, ou seja, q aponta para x
```

```
Logo:
```

```
*p, que é x, vale 30
```

```
*q, que é igual a *p, que é x, vale 30
```

```
*r é y, que vale 21
```

```
x vale 30
```

```
y vale 21
```

```
Logo, a resposta certa é 30 30 21 30 21
```

### 2 Questão

As structs (estruturas) são utilizadas para modelar os nodos de estruturas dinâmicas como, por exemplo, as listas encadeadas, seja o seguinte exemplo de nodo de uma lista de produtos:

```
struct nodo{  
    float valor;  
    string produto;  
    nodo * proximo;  
};
```

Suponha que um determinado ponteiro **pt** esteja apontando para um nodo desta lista, e que se queira alterar o conteúdo do campo **valor** deste nodo, que está sendo apontado por **pt**, para 5.60. Marque a alternativa que corretamente possibilita esta operação:

- ☐ `pt->5.60;`  
☒ `pt->valor=5.60;`  
☐ `pt->proximo.valor=5.60;`  
☐ `pt.valor->5.60;`  
☐ `pt->proximo->valor=5.60;`

Respondido em 20/04/2022 08:31:53

#### Explicação:

O código será:

```
pt->valor=5.60;
```

### 3 Questão

Qual é o resultado do código abaixo:

```
int a = 10;
```

```
int *p = &a;
```

```
cout<< &p << endl;
```

- ☐ O conteúdo da variável a será impresso  
☐ O endereço da variável a será impresso  
☐ Nenhuma das opções anteriores  
☐ O conteúdo da variável p será impresso  
☒ O endereço da variável p será impresso

Respondido em 20/04/2022 08:31:59

#### Explicação:

Analisando o trecho de código:

Linha 1: A variável a recebeu o valor 10

Linha 2: p é declarado como ponteiro para inteiro e recebe, na declaração, o endereço de a.

Linha 3: É impresso o endereço do ponteiro p.

Opções:

&gt;&gt; O endereço da variável a será impresso

Explicação: Falso. O endereço de a é p ou &amp;a.

&gt;&gt; O endereço da variável p será impresso

Explicação: Correto. Usou o operador &amp; antes de p.

&gt;&gt; O conteúdo da variável a será impresso

Explicação: Falso. Para imprimir a temos que fazer cout &lt;&lt; a; ou fazer cout &lt;&lt; \*p;

O conteúdo da variável p será impresso

Explicação: Falso. É impresso o endereço de p, pois se usou &amp;.

Nenhuma das opções anteriores

Explicação: Falso. Basta ver as opções anteriores.

### 4 Questão

Seja a seguinte sequência de instruções em um programa C++: `int *pti; int i = 10; pti = &i; Qual afirmativa e falsa?`

- ☐ \*pti é igual a 10
- ☐ pti armazena o endereço de i
- ☐ ao se alterar o valor de i, \*pti será modificado
- ☒ pti é igual a 10
- ☐ ao se executar \*pti = 20; i passará a ter o valor 20

Respondido em 20/04/2022 08:32:04

Explicação:

Analisando cada opção :

☒ pti é igual a 10

Explicação : A afirmativa está incorreta porque \*pti é igual 10 e não pti é igual a 10. Note que pti é o endereço de i. Logo, a afirmativa deve ser marcada.

☒ pti armazena o endereço de i

Explicação : pti é um ponteiro que recebeu o endereço (&) de i. Logo, a afirmativa está correta e não deve ser marcada.

☒ \*pti é igual a 10

Explicação : Afirmativa correta e não deve ser marcada, visto que \* é o operador de indireção. Como pti aponta para i, logo \*pti é i, que vale 10.

☒ ao se executar \*pti = 20; i passará a ter o valor 20

Explicação : Afirmativa correta e não deve ser marcada, visto que \*pti acessa a área apontada por i. Se \*pti recebeu 20 significa que i recebeu 20.

☒ ao se alterar o valor de i, \*pti será modificado

Explicação : Afirmativa correta e não deve ser marcada, visto que pti aponta para i e dessa forma, \*pti é i.

## 5 Questão

Montei uma biblioteca popular que aceita doações sem limites. Marque a opção que especifica o tipo de lista e o tipo de alocação de memória mais adequados para este problema.

- ☐ pilha dinâmica e alocação dinâmica de memória
- ☐ fila encadeada e alocação dinâmica de memória
- ☐ lista sequencial e alocação dinâmica de memória
- ☐ fila dinâmica e alocação dinâmica de memória
- ☒ lista encadeada e alocação dinâmica de memória

Respondido em 20/04/2022 08:32:09

[Gabarito](#)  
[Comentado](#)

## 6 Questão

Sobre listas encadeadas, é INCORRETO afirmar que:

- ☐ são acessadas pelo primeiro nodo da lista;
- ☐ pilhas e filas são versões limitadas de listas encadeadas, pois as inserções e remoções não ocorrem em qualquer parte.
- ☐ o final da lista faz uma referência para NULL;
- ☒ possuem tamanho fixo;
- ☐ a memória é alocada dinamicamente;

Respondido em 20/04/2022 08:32:14

Explicação:

Uma lista encadeada não tem tamanho fixo, pois usa-se alocação e desalocação dinâmica de memória. As demais afirmativas estão corretas.

## 7 Questão

Uma lista linear implementada através de encadeamento deve apresentar obrigatoriamente:

- i) um ponteiro para o primeiro nó da lista;
- ii) encadeamento entre os nós, através de algum campo de ligação;
- iii) não deve haver uma indicação de final da lista;
- iv) um ponteiro para o final da lista.

Levando em consideração as afirmações apresentadas, qual das alternativas é a correta?

- ☐ as afirmações i, ii e iii estão corretas.
- ☐ somente a afirmação i está correta.
- ☐ somente a afirmação iv está correta.
- ☒ as afirmações i e ii estão corretas.
- ☐ as afirmações ii e iv estão corretas.

Respondido em 20/04/2022 08:32:19

## 8 Questão

Em uma lista linear simplesmente encadeada.

- ☐ O ponteiro do "último" nó não é NULL, mas sim aponta de volta para o "primeiro" nó da lista.
- ☐ O ponteiro do "primeiro" nó não é NULL, mas sim aponta de volta para o "primeiro" nó da lista, formando um ciclo.
- ☒ Cada nó possui um só ponteiro que referencia o próximo nó da lista.
- ☐ Cada ponteiro possui um só endereço que referencia o primeiro nó da lista.
- ☐ Além do campo relativo ao dado, cada nó possui dois ponteiros,

Respondido em 20/04/2022 08:32:25

[Gabarito](#)  
[Comentado](#)

Col@bore

Antes de finalizar, clique aqui para dar a sua opinião sobre as questões deste simulado.

[Sugira!](#) [Sinalize!](#) [Construa!](#)