

## Lista de Exercícios 1

- 1) Analise o código fonte apresentado, considerando que será informado o valor 9 para a variável “a”. Assinale entre as alternativas a que representa o valor exibido como resultado nesta condição:

```
#include <iostream>
using namespace std;
int *p, *p2, a = 9;
main()
{
    p = &a;
    *p = 2 * *p;
    a += 3 + *p;
    p2 = p;
    cout << *p2 << endl;
}
```

- A) 21  
B) 12  
C) 30  
D) 39

- 2) Construa um programa que:
- Aloque dinamicamente um vetor de 10 posições;
  - Gravar as 10 posições do vetor com valores aleatório, intervalo de 20 a 50;
  - Mostre o vetor;
  - Libere a memória alocada.
- 3) Construa um programa que apresente um menu com as seguintes opções:
- Gerar vetor - alocar dinamicamente um vetor de 10 posições com valores aleatórios no intervalo de 20 a 50;
  - Mostrar vetor – mostrar os elementos do vetor;
  - Sair do programa.

### Observações:

- Antes de apresentar o menu, inicialize o vetor como NULL (`int *vetor = NULL;`)
- Para gerar um novo vetor, caso o vetor já tenha sido alocado, deve-se liberar memória antes de alocar um novo:  

```
if (vetor != NULL)
    // liberar memória...
```
- Na opção mostrar vetor, o vetor não pode ser nulo.

- 4) Crie um programa que declare uma estrutura (registro) para o cadastro de clientes.
- Deverão ser armazenados, para cada cliente: CPF, nome completo e ano de nascimento.
  - Ao início do programa, o usuário deverá informar o número de clientes que serão armazenados
  - O programa deverá alocar dinamicamente a quantidade necessária de memória para armazenar os registros dos clientes.
  - O programa deverá pedir ao usuário que entre com as informações clientes.
  - Ao final, mostrar os dados armazenados e liberar a memória alocada.

- 5) Faça um programa que crie dinamicamente um vetor de  $n$  elementos e passe esse vetor para uma função que vai ler os elementos desse vetor. Depois, no programa principal (main), o vetor preenchido deve ser impresso. Além disso, antes de finalizar o programa, deve-se liberar a área de memória alocada.
- 6) Crie uma função para calcular a soma de duas matrizes  $M \times N$  de números reais alocadas dinamicamente. A função deve retornar a matriz com a soma realizada. A implementação deste programa deve considerar as dimensões fornecidas pelo usuário. Além disso, antes de finalizar o programa, deve-se liberar a área de memória alocada.