

## LISTA - ALOCAÇÃO DINÂMICA

1. Explique a diferença entre alocação estática, alocação automática e alocação dinâmica em C. Dê exemplos de código para cada uma.

2. Analise o seguinte código:

```
1. int *p = (int*) malloc(5 * sizeof(int));
2. for (int i = 0; i < 5; i++) {
3.     p[i] = i * 2;
4. }
5. free(p);
6. printf("%d", p[2]);
```

a) O código compila?

b) O que acontece na execução? Justifique.

3. Qual a diferença entre malloc, calloc e realloc em C? Cite um caso de uso para cada função.

4. Dado o código abaixo, identifique e corrija o erro:

```
1. int *p;
2. for (int i = 0; i < 10; i++) {
3.     p[i] = i;
4. }
```

5. Escreva um programa em C que aloque dinamicamente um vetor de n números inteiros (informado pelo usuário), leia os valores e calcule a média.

6. Implemente um programa que comece com capacidade para armazenar 2 valores float. Sempre que a capacidade for excedida, o programa deve dobrar o espaço do vetor usando realloc.

7. Explique por que é importante utilizar free() em programas que utilizam malloc ou calloc. O que pode acontecer se esquecermos de liberar a memória? Demonstre com um exemplo em C.

**8. Um vetor dinâmico que dobra sua capacidade sempre que atinge o limite tem complexidade média de inserção O(1). Explique por que isso ocorre.**

**9. Crie uma struct chamada Aluno com os campos: nome, idade e nota. Faça um programa que aloque dinamicamente memória para armazenar um vetor de alunos e permita cadastrar e imprimir seus dados.**

**10. Implemente em C um vetor dinâmico genérico que suporte:**

- **inicialização com capacidade n;**
- **inserção de elementos;**
- **remoção de elementos;**
- **redimensionamento automático com realloc.**