

## Exercícios com TAD e Alocação Dinâmica de Memória

1. Construa uma TAD Pilha com alocação dinâmica de memória e que contenham todas as funções vistas em aula. Faça uma função main() que:

- Insira 10 números quaisquer
- A cada inserção, exiba o Início da Pilha
- Remova os 10 elementos da Pilha
- A cada remoção, exiba o topo da Pilha

2. Construa uma TAD Fila com alocação dinâmica de memória e que contenham todas as funções vistas em aula. Faça uma função main() que:

- Insira 10 números quaisquer
- A cada inserção, exiba o Início da Fila
- Remova os 10 elementos da Fila
- A cada remoção, exiba o topo da Fila

3. Faça um programa que leia o tamanho de um vetor de inteiros e reserve dinamicamente memória para esse vetor. Em seguida, leia os elementos desse vetor, imprima o vetor lido e mostre o resultado da soma dos números ímpares presentes no vetor.

4. Considere a seguinte estrutura:

```
typedef struct aluno{  
    char nome[30];  
    float media;  
    int faltas;  
}Aluno;
```

Faça um programa que leia informações de n alunos em um vetor alocado dinamicamente. Em seguida, imprima as informações lidas na ordem decrescente das médias dos alunos.

5. Construa um programa para ler N números a serem fornecidos pelo usuário. Todavia, a quantidade N de números não será fornecida. Desta forma, deverá alocado dinamicamente um vetor com 1 posição e a cada nova inserção o vetor deverá ser redimensionado para guardar o novo valor fornecido.

● Observação:

– Não é possível redimensionar a memória alocada. Assim, a cada nova posição a ser alocada deverá ser criado um novo vetor com 1 posição a mais e o conteúdo do vetor anterior deverá ser copiado para o novo vetor.