

LABORATÓRIO 15

ALOCAÇÃO DINÂMICA DE MEMÓRIA

EXERCÍCIOS DE REVISÃO

VOCÊ DEVE ACOMPANHAR PARA OBTER INFORMAÇÕES COMPLEMENTARES

- Peça ao usuário para digitar um número inteiro positivo e então crie um vetor dinâmico com essa quantidade de inteiros.

```
Digite o tamanho do vetor: 5
Criando vetor para 5 inteiros...
```

- Defina um registro que descreva um **peixe**. O registro deve incluir o **tipo** (string), o **peso** (ponto-flutuante) e o **comprimento** (inteiro) do peixe.

```
struct peixe
{
    char tipo[20];
    float peso;
    unsigned comp;
};
```

Em seguida mostre:

- Como criar uma variável de tipo **peixe**
 - Como alocar dinamicamente um registro de tipo **peixe**.
- Escreva o código para criar um vetor dinâmico de peixes. O tamanho do vetor deve ser definido pelo usuário. Mostre como ler e exibir o peso do segundo peixe.

```
Tamanho do vetor: 10
Tipo: Piaba
Peso: 6.7
Comprimento: 8

O peso do peixe é 6.7 gramas.
```

- Considerando o vetor dinâmico de peixes da questão anterior, crie uma função para receber um vetor de peixes. A função deve mostrar o peso do primeiro peixe.

EXERCÍCIOS DE FIXAÇÃO

VOCÊ DEVE FAZER OS EXERCÍCIOS PARA FIXAR O CONTEÚDO

1. Crie um programa que peça ao usuário para digitar o número de alunos em uma turma. O programa deve usar essa informação para criar um vetor dinâmico que armazene as notas finais desses alunos. Peça ao usuário para entrar com a nota de dois alunos e em seguida mostre essas notas usando cout.

```
Digite o número de alunos (mínimo 2): 10
Digite a nota de dois alunos:
4.5 8.9
As notas digitadas foram 4.5 e 8.9
```

2. Defina o registro **balao** como mostrado abaixo. Construa um programa para alocar dinamicamente uma variável do tipo **balao**. Peça ao usuário para entrar com valores para cada um dos membros e em seguida exiba o conteúdo do registro.

```
struct balao
{
    float diametro; // diâmetro em metros
    char marca[20]; // nome da marca
    int modelo; // número do modelo
};
```

3. Construa um registro para guardar informações sobre um **carro**. Um carro deve ter um modelo, ano de fabricação e preço. Em seguida construa um vetor estático de 10 carros inicializando os dois primeiros carros respectivamente para "Vectra", 2009, R\$58.000,00 e "Polo", 2008, R\$45.000,00. Use um ponteiro para apontar para o segundo carro e exibir seus dados.
4. Repita o exercício anterior criando um vetor dinâmico de carros. Ao invés de inicializar o vetor com valores predefinidos, peça ao usuário para digitar os dados de dois carros. Use uma função para receber o vetor de carros e exibir o valor total dos carros.

```
Entre com os dados de 2 carros:
Agile 2013 27500
Fusion 2017 112300
O valor total é R$139.800
```

5. Construa um vetor dinâmico de alunos. O registro **aluno** deve ser composto por nome (ou matrícula), código da disciplina (número inteiro sem sinal), e situação da disciplina. A situação da disciplina deve ser uma enumeração com os valores: Aprovado, Trancado, Reprovado. Peça ao usuário para digitar o número de alunos do vetor e em seguida leia os dados do primeiro aluno. Para finalizar mostre os dados do primeiro aluno usando uma função que recebe um ponteiro para **aluno**.

EXERCÍCIOS DE APRENDIZAGEM

VOCÊ DEVE ESCRER PROGRAMAS PARA REALMENTE APRENDER

- As instruções abaixo resultam em um código válido? Explique o porquê.

```
float peso;    Não é válido, pois a variável peso não pode ser deletada
peso = 30;      com o operador delete já que ela foi iniciada automaticamente
cout << peso;  e não manualmente, assim sendo de responsabilidade do
delete peso;  programa.
```

- Declare um ponteiro para inteiro, aloque memória dinamicamente para ele e armazene o número 100 nessa memória. Mostre o conteúdo apontado. Peça que o usuário digite um novo número inteiro e armazene-o na memória previamente alocada. Libere o espaço alocado dinamicamente ao final do programa.

```
Conteúdo armazenado: 100
Digite novo valor para esse bloco de memória: 80
```

- Inicie o programa perguntando ao usuário quantos inteiros ele deseja armazenar em um vetor. Use a informação digitada para criar um vetor dinâmico com o espaço necessário para armazenar a quantidade de inteiros desejada. Depois disso, deixe que o próprio usuário preencha o vetor, utilizando o tamanho do vetor como condição de parada de um laço for. Mostre o vetor que foi preenchido através de outro laço e libere o espaço alocado dinamicamente ao final do programa.

```
Quantos valores deseja guardar? 5
Quais os valores? 34 25 18 60 41
Os valores 34, 25, 18, 60 e 41 foram armazenados.
```

Sugestão: pesquise como usar um laço for para percorrer um vetor.

- Crie um registro "Local" com os campos nome, país e continente. Pergunte ao usuário quantos locais ele quer visitar nas próximas férias e crie um vetor de locais alocando dinamicamente o espaço de acordo com quantos locais ele quer visitar. Use um laço for para armazenar as informações dos locais que o usuário deseja visitar, e depois do armazenamento mostre os locais que ele escolheu. Libere o espaço alocado dinamicamente ao final do programa.
- Defina um registro ASCII que armazena um caractere e um valor inteiro associado. Crie uma função que recebe um valor inteiro e um caractere, e retorna o endereço de um elemento do tipo ASCII, alocado dinamicamente na memória. O programa principal deve chamar a função passando valores lidos do usuário, receber o retorno em um ponteiro, exibir os valores de retorno e deletar a memória que foi alocada dentro da função.

Dica: funções que retornam memória alocada são perigosas. É fácil esquecer de guardar o endereço de retorno para dar o delete.