

## Lista 3

Prof. Mateus Ferreira Satler

DECSI – ICEA - UFOP

### PARTE I – PONTEIROS E ALOCAÇÃO DINÂMICA DE MEMÓRIA

1. Faça um programa para alocar espaço em memória para 10 inteiros e peça ao usuário para digitar 10 valores. Posteriormente, imprima os seus respectivos endereço de memórias e o seu conteúdo.
2. Faça um programa que leia uma sequência de 5 números reais do teclado usando ponteiros em lugar de vetores estáticos. Determine o maior e o menor valor.
3. Faça um programa para ler uma sequência de números inteiros do teclado. A leitura deve parar quando o usuário digitar **zero**. Depois, exiba os números digitados na tela. **Dica:** comece alocando uma certa quantidade de espaço. Caso o espaço acabe, alocue mais espaço e copie o conteúdo já digitado para este novo espaço.
4. Faça um programa que leia n inteiros (definidos pelo usuário) armazenando-os em uma memória alocada dinamicamente. Em seguida, imprima na tela a quantidade dos n números que são pares e a quantidade que são ímpares.
5. Escreva um programa que lê primeiros os 6 números gerados pela loteria na noite de sábado na TV e depois lê os 6 números jogados por um apostador. Então, o programa compara quantos números o jogador acertou. Em seguida, ele aloca espaço para um vetor de tamanho igual a quantidade de números corretos e guarda os números corretos nesse vetor. Finalmente, o programa exibe os números sorteados (6 números) e os números que o apostador acertou.

### PARTE II – ESTRUTURAS

6. Defina uma estrutura para representar o peso e a altura de uma pessoa. Considere a inicialização de variáveis abaixo:  
  
    pessoa joao, maria;  
    joao.altura = 1.90; joao.peso = 98;  
    maria.altura = 1.50; maria.peso = 55;  
  
    a) Escreva uma instrução que atribua 1.78 à altura de joao.  
    b) Escreva uma instrução que atribua 75 ao peso de maria.  
    c) Escreva um conjunto de instruções para imprimir a média das alturas e a média dos pesos de joao e maria.
7. Faça um programa (função principal) para leitura, via teclado, dos dados de um aluno. Os dados a serem guardados na estrutura aluno são os seguintes: nome, curso, idade. Ao final, imprima estas informações na tela.

8. Considere a mesma estrutura definida no exercício anterior. Acrescente à estrutura um vetor com as notas das três provas feitas pelo aluno, calcule a sua média e diga se ele foi aprovado ou não (media  $\geq$  60). Para o cálculo da média e leitura das notas, utilize o comando for (que “varrerá” o vetor da estrutura).
9. Crie uma estrutura chamada ponto contendo apenas as coordenadas x e y (inteiros) do ponto. Declare 2 pontos, leia as coordenadas x e y de cada um e calcule a distância entre eles. Apresente no final a distância entre os dois pontos.
10. Faça um programa que armazene as informações de 11 jogadores de um time de futebol. Cada jogador é identificado pelo número da sua camisa, pelo seu peso, altura e pela inicial do seu nome. O programa deverá ler todas as informações e imprimir ao final a lista completa de jogadores com todas as informações lidas, a inicial do jogador mais baixo e o número do mais pesado.
11. Faça um programa para fazer o cadastro dos veículos de uma empresa. Poderão ser cadastrados no máximo 100 veículos e as informações a serem armazenadas são as seguintes: nome do condutor, placa do veículo e cor do veículo. Armazene também o turno que os veículos são armazenados como sendo m (manhã), t (tarde), n (noite) e i (dia inteiro). Informe ao final a lista completa dos veículos cadastro com todas as informações que forem cadastradas. Utilize procedimentos para fazer a leitura e a escrita dos dados.
12. Considere a seguinte estrutura:

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

Faça um programa que leia informações de n alunos em um vetor alocado dinamicamente. Pergunte ao usuário se ele deseja incluir um novo aluno. Em caso afirmativo, crie espaço para armazenar este novo aluno. Em seguida, imprima as informações lidas.