



UNIVERSIDADE FEDERAL DE UBERLÂNDIA

Faculdade de Computação

Avenida João Naves de Ávila, 2121, Bloco 1B - Bairro Santa Mônica, Uberlândia/MG, CEP 38400-902

Telefone: +55 (34) 3239-4218 - www.facom.ufu.br - cocom@ufu.br



Bacharelado em Ciência da Computação

Bacharelado em Sistemas de Informação

Disciplina: Programação Procedimental - PP [GBC014/GSI002]

Prof. Me. Claudiney R. Tinoco

Lab06 - Estruturas

1. Escreva um trecho de código para fazer a criação dos novos tipos de dados conforme solicitado abaixo:
 - Horário: composto de hora, minutos e segundos.
 - Data: composto de dia, mês e ano.
 - Compromisso: composto de uma data, horário e texto que descreve o compromisso.
2. Implemente um programa que leia o nome, a idade e o endereço de uma pessoa e armazene os dados em uma estrutura.
3. Construa uma estrutura aluno com nome, número de matrícula e curso. Leia do usuário a informação de 5 alunos, armazene em um vetor dessa estrutura e imprima os dados na tela.
4. Crie uma estrutura representando os alunos de um determinado curso. A estrutura deve conter a matrícula do aluno, nome, nota da primeira prova, nota da segunda prova e nota da terceira prova.
 - a) Permita ao usuário entrar com os dados de 5 alunos;
 - b) Encontre o aluno com maior nota da primeira prova;
 - c) Encontre o aluno com maior média geral;
 - d) Encontre o aluno com menor média geral;
 - e) Para cada aluno diga se ele foi aprovado ou reprovado, considerando o valor 6 para aprovação.

5. Considerando a estrutura

```
struct vetor {  
    float a;  
    float b;  
    float c;  
};
```

para representar um vetor no R_3 , implemente um programa que calcule a soma de dois vetores.

6. Faça um programa que realize a leitura dos seguintes dados relativos a um conjunto de alunos: Matrícula, Nome, Código da Disciplina, Nota1 e Nota2. Considere uma turma de até 10 alunos. Após ler todos os dados digitados, e depois de armazená-los em um vetor de estrutura, exibir na tela a listagem final dos alunos com as suas respectivas médias finais (use uma média ponderada: Nota1 com peso = 1.0 e Nota2 com peso = 2.0).

7. Faça um programa que converta coordenadas polares para cartesianas:
- Crie e leia um ponto em coordenada polar, composto por raio (r) e argumento (a) em radianos;
 - Crie outro ponto, agora em coordenada cartesiana, composto por x e y , sabendo que $x = r * \cos(a)$ e $y = r * \sin(a)$.

No programa principal, leia um ponto em coordenada polar e mostre as coordenadas do ponto gerado no plano cartesiano.

8. Faça um programa que armazene em um registro de dados (estrutura composta) os dados de um funcionário de uma empresa, compostos de: Nome, Idade, Sexo (M/F), CPF, Data de Nascimento, Código do Setor onde trabalha (0-99), Cargo que ocupa (string de até 30 caracteres) e Salário. Os dados devem ser digitados pelo usuário, armazenados na estrutura e exibidos na tela.
9. Faça um programa que faça operações simples de números complexos:
- Crie e leia dois números complexos z e w , compostos por parte real e parte imaginária;
 - Apresente a soma, subtração e produto entre z e w , nessa ordem, bem como o módulo de ambos.
10. Utilizando uma estrutura, faça um programa que permita a entrada de nome, endereço e telefone de 5 pessoas e os imprima em ordem alfabética.
11. Faça um programa que leia os dados de 10 alunos (Nome, matrícula, Média Final), armazenando em um vetor. Uma vez lidos os dados, divida estes dados em 2 novos vetores, o vetor dos aprovados e o vetor dos reprovados, considerando a média mínima para a aprovação como sendo 5.0. Exibir na tela os dados do vetor de aprovados, seguido dos dados do vetor de reprovados.
12. Escolha um jogo de cartas, baseado em um “baralho tradicional” (cada carta tem seu naipe e seu valor). Implemente a parte de distribuição (sorteio) de cartas para 2 jogadores, considerando que cada jogador irá receber 3 cartas. Exiba na tela as cartas que cada um dos jogadores recebeu.
13. Peça ao usuário para digitar seus dados pessoais (Nome, Endereço, Data de Nascimento, Cidade, CEP, e-mail), verifique se as informações de Data de Nascimento, CEP e e-mail fazem sentido, e mostre ao usuário as informações, se estão todas corretas, ou mostre que alguma informação estava errada.
14. Faça um programa que leia um vetor com os dados de 5 carros: marca (máximo 15 letras), ano e preço. Leia um valor p e mostre as informações de todos os carros com preço menor que p . Repita este processo até que seja lido um valor $p = 0$.
15. Faça um programa que leia um vetor com dados de 5 livros: título (máximo 30 letras), autor (máximo 15 letras) e ano. Procure um livro por título, perguntando ao usuário qual título deseja buscar. Mostre os dados de todos os livros encontrados.
16. Faça um programa que seja uma agenda de compromissos e:
- Crie e leia um vetor de 5 estruturas de dados com: compromisso (máximo 60 letras) e data. A data deve ser outra estrutura de dados contendo dia, mês e ano;
 - Leia dois inteiros M e A e mostre todos os compromissos do mês M do ano A . Repita o procedimento até ler $M = 0$.

17. Faça um programa que controla o consumo de energia dos eletrodomésticos de sua casa e:
- Crie e leia 5 eletrodomésticos que contém nome (máximo 15 letras), potência (real, em kW) e tempo ativo por dia (real, em horas);
 - Leia um tempo t (em dias), calcule e mostre o consumo total na casa e o consumo relativo de cada eletrodoméstico (consumo/consumo total) nesse período de tempo. Apresente este último dado em porcentagem.
18. Faça um programa que controle o fluxo de voos nos aeroportos de um país. Com $V = 5$ (voos) e $A = 5$ (aeroportos) e:
- Crie e leia um vetor de voos, sendo que cada voo contém um código de aeroporto de origem e um de destino;
 - Crie um vetor de aeroportos, sendo que cada aeroporto contém seu código, quantidade de voos que saem e quantidade de voos que chegam.
- Nota:** Cada aeroporto é identificado por um código inteiro entre 0 e $(A-1)$. Não aceite aeroportos de código inexistente.
19. Faça um programa que armazena filmes produzidos por vários diretores e:
- Crie e leia um vetor de 5 diretores, cada um contendo nome (máximo 20 letras), quantidade de filmes e filmes. O membro filmes é um vetor, que deve ser criado após ter lido quantidade de filmes. Cada filme é composto por nome, ano e duração;
 - Procure um diretor por nome, mostrando todos os filmes que ele já produziu. Repita o processo até digitar uma string vazia.
20. Escreva um programa que receba dois structs do tipo “dma”, cada um representando uma data válida, e calcule o número de dias que decorreram entre as duas datas.

```
struct dma {  
    int dia;  
    int mes;  
    int ano;  
};
```