



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: Algoritmos e Estruturas de Dados 1 – AED1 [GBS024/GSI006]

Prof. Me. Claudiney R. Tinoco

1ª Lista de Exercícios – Revisão C: Fundamentos e Tipos de Dados

- 1) Três amigos jogaram na loteria. Caso eles ganhem, o prêmio deve ser repartido proporcionalmente ao valor que cada deu para a realização da aposta. Faça um programa que lê quanto cada apostador investiu, lê o valor do prêmio, e escreve quanto cada um ganharia.

Exemplo de saída:

```
<< Loteria >>
Bolão jogador 1 R$: 3
Bolão jogador 2 R$: 3
Bolão jogador 3 R$: 4
Informe o valor do prêmio: 6000.0
Jogador 1 receberá R$: 1800.0
Jogador 2 receberá R$: 1800.0
Jogador 3 receberá R$: 2400.0
```

- 2) Escreva um programa que leia as coordenadas x e y de pontos no R2 e calcule sua distância da origem (0,0). Obs.: faça #include <math.h> e use a função sqrt() para calcular a raiz de um número.

Exemplo de saída:

```
<< Distância >>
Entre com a coordenada x: 3
Entre com a coordenada y: 4
A distância entre os pontos (3,4) e (0,0) eh 5.0
```

- 3) Faça um conversor de número binário para decimal (inteiro). O número deve ter 4 bits, sendo o bit mais significativo digitado primeiro. Mostrar na resposta o número binário e o número decimal.

Exemplo de saída:

```
== Conversor de números binários ==
Digite o 1o. bit: 1
Digite o 2o. bit: 0
Digite o 3o. bit: 1
Digite o 4o. bit: 1
O número binário 1011 corresponde ao número decimal 11
```

Exemplo de como é feita a conversão. Lembre-se que para este exercício são necessários somente 4 bits

$$\begin{array}{cccccccccccc}
 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
 2^{10} \times 1 & 2^9 \times 1 & 2^8 \times 1 & 2^7 \times 1 & 2^6 \times 1 & 2^5 \times 0 & 2^4 \times 0 & 2^3 \times 0 & 2^2 \times 0 & 2^1 \times 0 & 2^0 \times 1 \\
 1024 & + & 512 & + & 256 & + & 128 & + & 64 & + & 0 & + & 0 & + & 0 & + & 0 & + & 0 & + & 0 & + & 1 \\
 & & & & & & & & & & = & & & & & & & & & & & 1985
 \end{array}$$

- 4) Faça um algoritmo que calcule o IMC de uma pessoa e mostre sua classificação de acordo com a tabela abaixo

$$IMC = \frac{massa (kg)}{altura(m)^2}$$

IMC	Classificação
< 18,5	Magreza
18,5 – 24,9	Saudável
25,0 – 29,9	Sobrepeso
30,0 – 34,9	Obesidade Grau I
35,0 – 39,9	Obesidade Grau II (severa)
≥ 40,0	Obesidade Grau III (morbida)

(Fonte: Wikipédia)

- 5) Escreva um programa que leia um número inteiro positivo n e em seguida imprima n linhas do chamado Triângulo de Floyd:

```

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 19 20 21

```

- 6) Faça um programa que mostre o valor de 1!, 2!, 3!, até 10!. Utilize somente variáveis inteiras

Exemplo de saída:

```

<< Fatorial >>
1! = 1
2! = 2
3! = 6
4! = 24
5! = 120
6! = 720
7! = 5040
8! = 40320
9! = 362880
10! = 3628800

```

- 7) Altere o programa anterior para calcular de 10! até 15!. É esperado que ocorra um erro. Explique o motivo do erro e qual a solução.

Exemplo de saída:

```
<< Fatorial >>
11 ! = 39 916 800
12 ! = 479 001 600
13 ! = 6 227 020 800
14 ! = 87 178 291 200
15 ! = 1 307 674 368 000
```

- 8) Calcular a média e o desvio padrão amostral de 5 números informados pelo usuário.

Média:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{x_1 + x_2 + \dots + x_n}{n}$$

Desvio-padrão (amostral, normalizado por N-1)

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$$

(Fonte: Wikipédia)

Exemplo de saída:

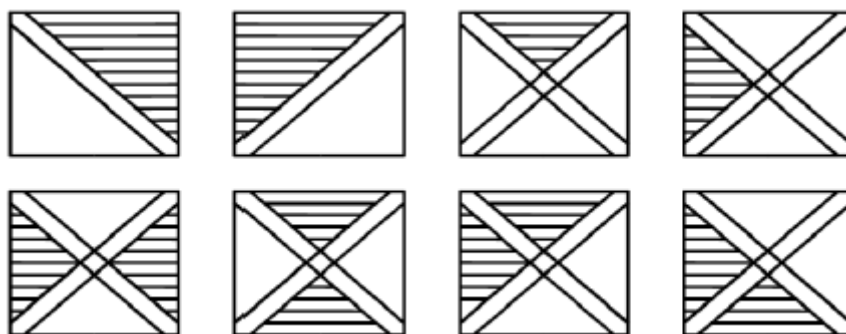
```
<< Média e desvio-padrão >>
Digite o valor 1: 50
Digite o valor 2: 60
Digite o valor 3: 70
Digite o valor 4: 80
Digite o valor 5: 60
A média é 64 e o desvio-padrão é 11.4017542509914
```

- 9) Faça um programa que leia o nome de uma pessoa e mostre esse nome em caixa alta (pense como fazer isso olhando a diferença entre os valores dos caracteres minúsculos e maiúsculos na tabela ASCII). Não usar as funções de string.h

Exemplo de saída:

```
<< Vetor de char>>
Digite um nome: Bruno
O nome digitado é: BRUNO
```

- 10) Faça um algoritmo que lê uma matriz 6x6 e soma os valores das partes hachuradas. Escreva a matriz e a soma das partes calculadas.



11) Fazer um programa para simular uma agenda de telefones. Para cada pessoa devem-se ter os seguintes dados:

- Nome
 - E-mail
 - Endereço (contendo campos para Rua, número, complemento, bairro, CEP, cidade, estado, país).
 - Telefone (contendo campo para DDD e número)
 - Data de aniversário (contendo campo para dia, mês, ano).
 - Observações: Uma linha (string) para alguma observação especial.
- a. Definir a estrutura acima.
 - b. Declarar a variável agenda (vetor) com capacidade de agendar até 100 nomes.
 - c. Definir um bloco de instruções busca por primeiro nome: Imprime os dados da pessoa com esse nome (se tiver mais de uma pessoa, imprime para todas).
 - d. Definir um bloco de instruções busca por mês de aniversário: Imprime os dados de todas as pessoas que fazem aniversário nesse mês.
 - e. Definir um bloco de instruções busca por dia e mês de aniversário: Imprime os dados de todas as pessoas que fazem aniversário nesse dia e mês.
 - f. Definir um bloco de instruções insere pessoa: Insere por ordem alfabética de nome.
 - g. Definir um bloco de instruções retira pessoa: Retira todos os dados dessa pessoa e desloca todos os elementos seguintes do vetor para a posição anterior.
 - h. Definir um bloco de instruções imprime agenda com as opções:
 - I. Imprime nome, telefone e e-mail.
 - II. Imprime todos os dados.
 - i. O programa deve ter um menu principal oferecendo as opções acima.