

Estruturas de Dados I

Funções e Procedimentos  
(2ª Parte)

UNICID - 2016

1. Crie um algoritmo (usando subalgoritmos) em Linguagem C, que leia uma temperatura dada em Farenheit e converta-a para Celsius.

Sabemos que:

$$tc = 5*(tf-32)/9$$

2. Crie um subalgoritmo em Linguagem C, para ler três (3) valores a, b e c e ordená-los de forma crescente.

se  $(n1 < n2) \text{ e } (n2 < n3) \Rightarrow n1, n2, n3$   
se  $(n1 < n3) \text{ e } (n3 < n2) \Rightarrow n1, n3, n2$   
se  $(n2 < n1) \text{ e } (n1 < n3) \Rightarrow n2, n1, n3$   
se  $(n2 < n3) \text{ e } (n3 < n1) \Rightarrow n2, n3, n1$   
se  $(n3 < n1) \text{ e } (n1 < n2) \Rightarrow n3, n1, n2$   
se  $(n3 < n2) \text{ e } (n2 < n1) \Rightarrow n3, n2, n1$

3. Crie uma subrotina em Linguagem C que cheque se três valores inteiros distintos A, B e C formam um triângulo e classifiquem os:

OBS: A, B e C formam um triângulo  $\Leftrightarrow$

$A < (B + C)$  e  $B < (A + C)$  e  $C < (A + B)$

Se  $(A = B)$  e  $(B = C) \Rightarrow$  Triângulo Equilátero

Se  $(A = B)$  ou  $(B = C)$  ou  $(A = C) \Rightarrow$  Triângulo Isósceles

Se  $(A \neq B)$  e  $(B \neq C)$  e  $(A \neq C) \Rightarrow$  Triângulo Escaleno

```
#include <stdio.h>

void triangulo(int a, int b, int c){
    if((a < (b + c)) && (b < (a+c)) && c < (a+b)){
        //a, b e c formam um triângulo
        printf("\n\n%d,%d e %d formam um triangulo: ",a,b,c);
        //Classificação do Triângulo
        if((a==b) && (b==c))
            printf("\nEquilatero");
        else
            if((a==b) || (b==c) || (a==c))
                printf("\nIsosceles");
            else
                printf("\nEscaleno");
    }
    else //a,b e c não formam um triângulo!
        printf("\n\n%d,%d e %d NAO formam um triangulo: ",a,b,c);
    getch();
}

main()
{
    int lado1,lado2,lado3;
    printf("TRIANGULOS\n\n");
    printf("\nDigite o 1o lado: ");
    scanf("%d",&lado1);
    printf("\nDigite o 2o lado: ");
    scanf("%d",&lado2);
    printf("\nDigite o 3o lado: ");
    scanf("%d",&lado3);
    triangulo(lado1,lado2,lado3);
}
```

3'. Crie uma subrotina que verifique se três valores  $a$ ,  $b$  e  $c$  podem formar um triângulo e calcule o valor da área do triângulo, se for possível, senão exibir uma mensagem dizendo que o trio não pode formar um triângulo:

$$A = \sqrt{p(p-a)(p-b)(p-c)}, \text{ sendo}$$
$$p = \frac{a+b+c}{2}$$

4. Crie um subalgoritmo em Linguagem C, para ler três (3) valores a, b e c reais e calcular e resolver a equação de segundo grau, usando a Fórmula de Báscara:

$$ax^2 + bx + c = 0$$

Sendo que  $\text{delta} = b^2 - 4 \cdot a \cdot c$

se  $\text{delta} > 0 \Rightarrow$  duas raízes:  $x_1 = (-b + \text{raiz}(\text{delta}))/2 \cdot a$

$$x_2 = (-b - \text{raiz}(\text{delta}))/2 \cdot a$$

Se  $\text{delta} = 0 \Rightarrow$  uma raiz igual a  $x_1 = -b/2 \cdot a$

Se  $\text{delta} < 0 \Rightarrow \nexists$  raízes reais

5. Faça um sub-algoritmo em Linguagem C, que recebe por parâmetro o tempo de duração de um intervalo de tempo expresso em segundos e retorne esse parâmetro em horas, minutos e segundos.

Ex:  $t = 9999 \text{ seg.}$

$1 \text{ hora} = 3600 \text{ seg.}$   
 $1 \text{ min} = 60 \text{ seg.}$

$$\begin{array}{r} 9999 \overline{) 3600} \\ - 7200 \\ \hline 2799 \end{array} \quad 2(h)$$

$$\begin{array}{r} 2799 \overline{) 60} \\ - 240 \\ \hline 399 \\ - 360 \\ \hline 39 \end{array} \quad 46(min)$$

$$\begin{array}{r} 39 \overline{) 60} \\ - 36 \\ \hline 24 \end{array} \quad 39(seg)$$

$2h \ 46min \ 39seg.$



6. Escreva uma função que recebe por parâmetro um valor inteiro e positivo N e retorna o valor de S.

$$S = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \dots + \frac{1}{N}.$$

N	i	soma
5	1	<del>0</del>
	2	$0 + 1/2 = 0,5$
	3	$0,5 + 1/3 = 0,83$
	4	$0,83 + 1/4 = 1,08$
	5	$1,08 + 1/5 = 1,28$

$$soma += 1/i;$$

$$soma = soma + 1/i;$$

$$N=5$$



$$soma = 2,28$$

7. Crie um subalgoritmo em Linguagem C que leia um número inteiro positivo  $n$  e calcule o seu fatorial:

Ex:  $n = 7 \Rightarrow 7! = 1 * 2 * 3 * 4 * 5 * 6 * 7 = 5040$

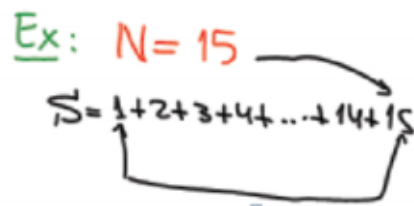
8. Crie um subalgoritmo que verifique se um número é primo ou não.

9. Crie um subalgoritmo em Linguagem C, que leia um número inteiro  $n$  e retorne na tela os seus divisores positivos:

Ex:  $n = 28 \Rightarrow$  deve retornar: 1, 2, 4, 7, 14, 28

10. Crie uma sub-rotina que leia um número inteiro e positivo  $N$  como parâmetro e retorne a soma dos números inteiros existentes entre o número 1 e  $N$ .

Ex:  $N = 15$

$$S = 1 + 2 + 3 + 4 + \dots + 14 + 15$$


11. Crie um subalgoritmo em Linguagem C, que leia um número inteiro  $n$  e diga se ele é perfeito ou não.

Obs: um  $n^\circ$  é perfeito se for igual à soma de seus divisores, exceto ele mesmo

Ex:  $n = 28$  é perfeito, pois  $1 + 2 + 4 + 7 + 14 = 28$

$n = 10$  não é perfeito, pois  $1 + 2 + 5 \neq 10$

## Passando Vetores para Funções

### Exemplo:

```
#include <stdio.h>
#include <conio.h>
double media( int[], int);
main(){
    int notas[20], i;
    double m;
    for(i=0; i<20; i++){
        scanf("%d",&notas[i]);
        if (notas[i] < 0) break;
    }
    m = media(notas, i);
    printf("\nA media das notas: %.1f",m);
    getch();
}
```

### • Implementação da função

```
double media( int lista[], int tamanho){
    double soma = 0;
    int i;
    for(i=0; i<tamanho; i++) {
        soma+=lista[i];
    }
    return(soma/tamanho);
}
```