

Linguagem C

Professor Alexandre Moreno

Procedimentos e Funções

Conforme vamos escrevendo o programa, é natural que o mesmo tenha um número expressivo de linhas. Desta forma, quebrar o programa e blocos facilitará a compreensão e possibilitará o reaproveitamento de partes já escritas.

DEFINIÇÃO: Bloco de comandos ao qual é atribuído um nome e pode ser chamada (executado) em qualquer parte do programa, quantas vezes forem necessárias. Tanto Procedimento quanto Funções podem fazer uso de parâmetros, mas não é uma obrigatoriedade.

VANTAGENS:

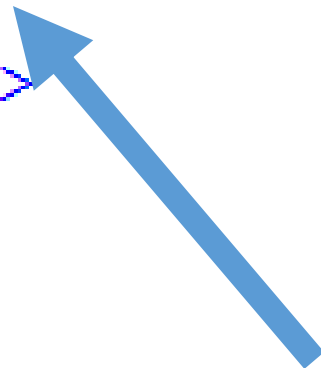
- Estruturação do programa;
- Reutilização de códigos.

Em geral, podemos dizer que **procedimento** executa uma tarefa, enquanto a **função** executa uma tarefa e sempre **RETORNA** um valor.

Toda linguagem possui procedimentos e funções pré-existentes, mas podemos (e até devemos) criar outras.

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    int n;
    printf("informe um valor: ");
    scanf("%d", &n);
    system("pause");
    return 0;
}
```



printf e **scanf** são exemplos de procedimento e função já existentes. Fazem parte da biblioteca **stdio.h**

```
#include <stdio.h>
#include <stdlib.h>
float a,b;
```

Variáveis **globais** (valem em todo programa)

```
void media() {
    float result;
    result = (a+b)/2;
    printf("media = %.2f\n", result);
}
```

Variável **local** (funcionam apenas neste bloco)

```
int main() {
    printf("Entre com o primeiro valor: ");
    scanf("%f", &a);
    printf("Entre com o segundo valor: ");
    scanf("%f", &b);
    media();
    system("pause");
    return(0);
}
```

```
#include <stdio.h>
#include <stdlib.h>
float a,b;
```

PROCEDIMENTO
sempre começa com a
palavra **void**.

```
void media() {
    float result;
    result = (a+b)/2;
    printf("media = %.2f\n",result);
}
```

```
int main() {
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    media();
    system("pause");
    return(0);
}
```

Chamada do
procedimento
media

Notem que **não** há
parâmetros no
procedimento
media()


```
#include <stdio.h>
#include <stdlib.h>
float a,b,result;
```

```
float media() {
    float result;
    result = (a+b)/2;
    return result;
}
```

FUNÇÃO sempre começa com o **tipo** que será retornado.

```
int main() {
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    result=media();
    printf("media = %.2f\n",result);
    system("pause");
    return(0);
}
```

Uma variável está recebendo o retorno da função `media()`

```
#include <stdio.h>
#include <stdlib.h>
float a,b;

float media() {
    float result;
    result = (a+b)/2;
    return result;
}

int main() {
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media());
    system("pause");
    return (0);
}
```

Apresentação direta
do retorno da função
media()

```
#include <stdio.h>
#include <stdlib.h>
float a,b,result;
```

```
float media() {
    float result;
    result = (a+b)/2;
    return result;
}
```

```
int main() {
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    result=media();
    printf("media = %.2f\n",result);
    system("pause");
    return(0);
}
```

Variável recebendo o retorna da função
media()

```
#include <stdio.h>
#include <stdlib.h>
float a,b;
```

```
float media() {
    float result;
    result = (a+b)/2;
    return result;
}
```

```
int main() {
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media());
    system("pause");
    return(0);
}
```

Apresentação **direta** do retorna da
função media()

Perceberam que tanto no procedimento quanto na função, a média foi calculada para as variáveis **a e b**?

```
void media() {  
    float result;  
    result = (a+b)/2;  
    printf("media = %.2f\n", result);  
}
```

```
float media() {  
    float result;  
    result = (a+b)/2;  
    return result;  
}
```

```
#include <stdio.h>
#include <stdlib.h>
float a,b,x,y;
float media() {
    float result;
    result = (a+b)/2;
    return result;
}
int main() {
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media());
    printf("Entre com o primeiro valor: ");
    scanf("%f",&x);
    printf("Entre com o segundo valor: ");
    scanf("%f",&y);
    printf("media = %.2f\n",media());

    system("pause");
    return(0);
}
```

Fazendo reuso da função `media()`, mas quais valores serão retornados?

```
Entre com o primeiro valor: 10
Entre com o segundo valor: 15
media = 12.50
```

```
Entre com o primeiro valor: 3
Entre com o segundo valor: 4
media = 12.50
```

```
Pressione qualquer tecla para continuar. . .
```

```
#include <stdio.h>
#include <stdlib.h>
float a,b,x,y;
float media() {
    float result;
    result = (a+b)/2;
    return result;
}

int main() {
    printf("Entre com o primeiro valor: ");
    scanf("%f", &a);
    printf("Entre com o segundo valor: ");
    scanf("%f", &b);
    printf("media = %.2f\n", media());

    printf("Entre com o primeiro valor: ");
    scanf("%f", &x);
    printf("Entre com o segundo valor: ");
    scanf("%f", &y);
    printf("media = %.2f\n", media());

    system("pause");
    return 0;
}
```

```
Entre com o primeiro valor: 10
Entre com o segundo valor: 15
media = 12.50
```

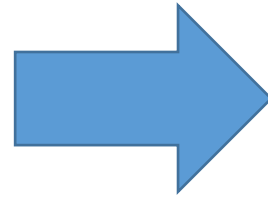
```
Entre com o primeiro valor: 3
Entre com o segundo valor: 4
media = 12.50
```

```
Pressione qualquer tecla para continuar. . .
```

O problema ocorre porque a função `media()`, sempre retorna o resultado calculado para $(a+b)/2$.

Solução: Criar PARÂMETROS

```
float media() {  
    float result;  
    result=(n1+n2)/2;  
    return result;  
}
```



```
float media() {  
    return n1+n2)/2;  
}
```

Otimizando a função

```
#include <stdio.h>
#include <stdlib.h>
float a,b,x,y;

float media(float n1, float n2){
    return (n1+n2)/2;
}

int main(){
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media(a,b));

    printf("Entre com o primeiro valor: ");
    scanf("%f",&x);
    printf("Entre com o segundo valor: ");
    scanf("%f",&y);
    printf("media = %.2f\n",media(x,y));

    system("pause");
    return(0);
}
```

Parâmetros formais (n1,n2),
ambos float

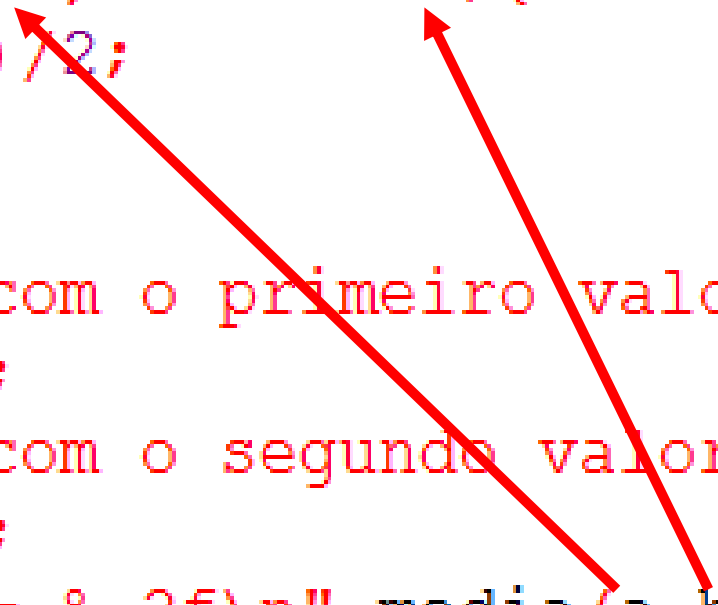
Parâmetros reais
(a,b), ambos float

Parâmetros reais
(x,y), ambos float


```
#include <stdio.h>
#include <stdlib.h>
float a,b,x,y;
```

O valor da variável **a** será passado para o parâmetro **n1**, e o valor da variável **b** para o parâmetro **n2**

```
float media(float n1, float n2) {
    return (n1+n2)/2;
}
int main() {
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media(a,b));
```



The diagram consists of two red arrows. The first arrow originates from the variable `a` in the function call `media(a,b)` within the `main` function and points to the parameter `n1` in the function definition `float media(float n1, float n2)`. The second arrow originates from the variable `b` in the function call `media(a,b)` and points to the parameter `n2` in the function definition.

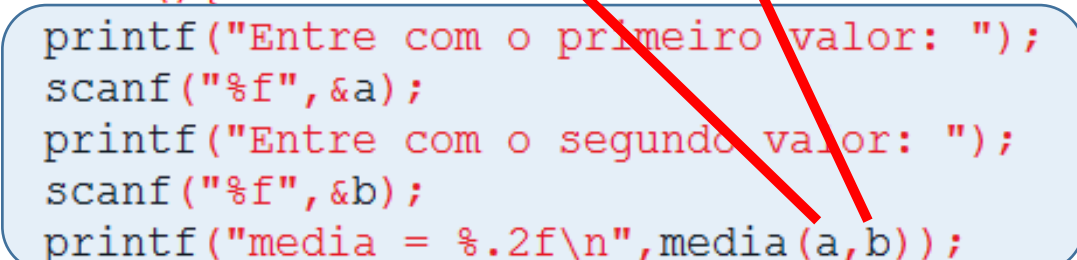
```
#include <stdio.h>
#include <stdlib.h>
float a,b,x,y;

float media(float n1, float n2){
    return (n1+n2)/2;
}

int main(){
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media(a,b));

    printf("Entre com o primeiro valor: ");
    scanf("%f",&x);
    printf("Entre com o segundo valor: ");
    scanf("%f",&y);
    printf("media = %.2f\n",media(x,y));

    system("pause");
    return 0;
}
```



The diagram illustrates the flow of data between the `main` function and the `media` function. Two red arrows originate from the `main` function's input section. The first arrow starts at the `scanf("%f",&a);` line and points to the `n1` parameter in the `media` function's signature. The second arrow starts at the `scanf("%f",&b);` line and points to the `n2` parameter in the `media` function's signature. This visualizes how the values of `a` and `b` are passed to the `media` function for calculation.

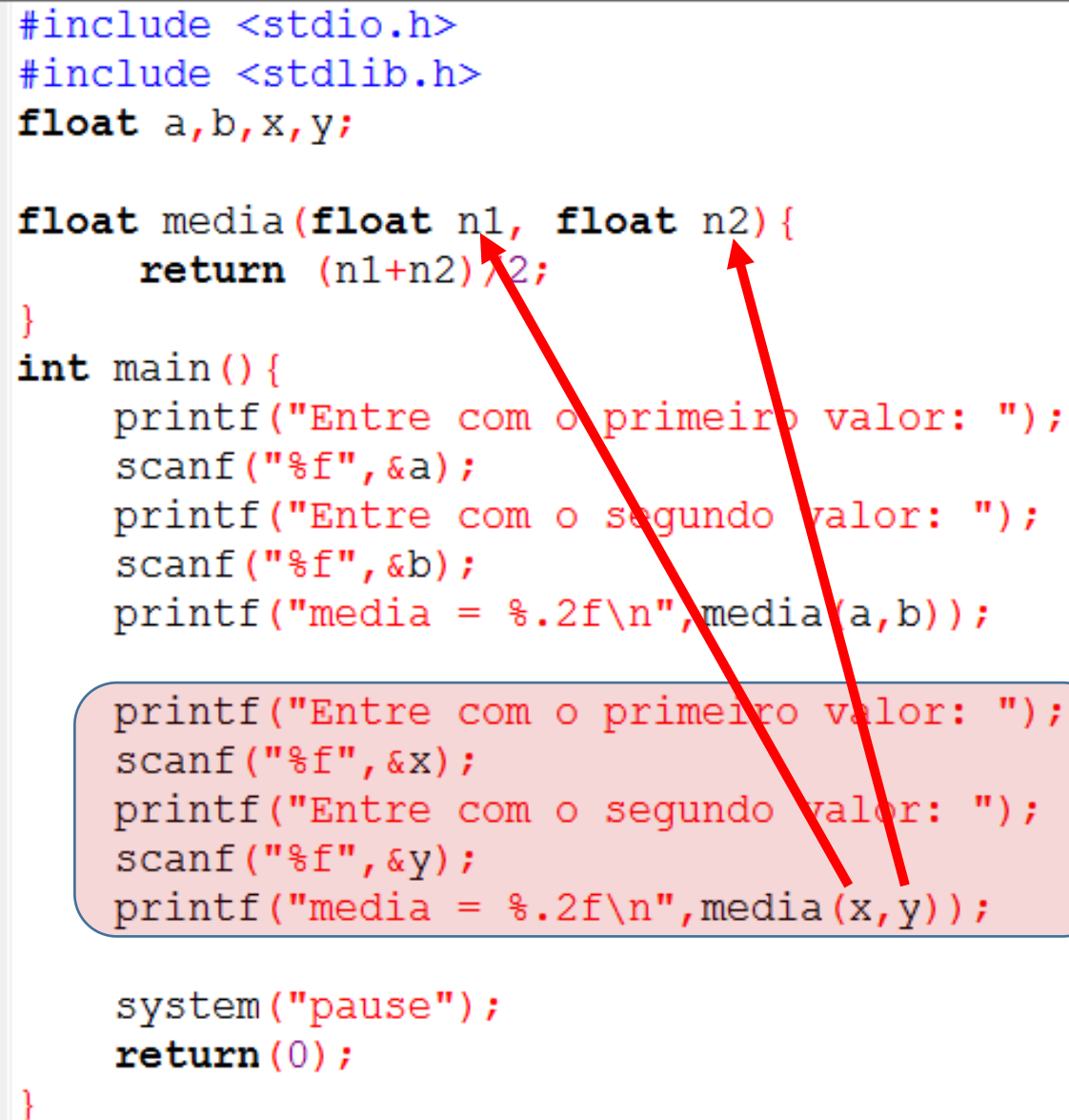
```
#include <stdio.h>
#include <stdlib.h>
float a,b,x,y;

float media(float n1, float n2){
    return (n1+n2)/2;
}

int main(){
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media(a,b));

    printf("Entre com o primeiro valor: ");
    scanf("%f",&x);
    printf("Entre com o segundo valor: ");
    scanf("%f",&y);
    printf("media = %.2f\n",media(x,y));

    system("pause");
    return 0;
}
```



```
#include <stdio.h>
#include <stdlib.h>
float a,b,x,y;

float media(float n1, float n2){
    return (n1+n2)/2;
}

int main(){
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media(a,b));

    printf("Entre com o primeiro valor: ");
    scanf("%f",&x);
    printf("Entre com o segundo valor: ");
    scanf("%f",&y);
    printf("media = %.2f\n",media(x,y));

    system("pause");
    return(0);
}
```

```
Entre com o primeiro valor: 10
Entre com o segundo valor: 15
media = 12.50
```

```
Entre com o primeiro valor: 3
Entre com o segundo valor: 5
media = 4.00
```

```
Pressione qualquer tecla para continuar. . .
```

PARÂMETROS servem tanto para procedimentos quanto funções. Uma vez estabelecido os parâmetros **formais** (nomenclatura, tipo e ordem), ao efetuar a chamada da sub-rotina (procedimento ou função), sempre teremos que passar os parâmetros **reais**, na mesma ordem e tipo aos quais foram definidos os parâmetros **formais**.

Existem dois tipos de passagem de parâmetro: Por valor e por referência. Os exemplos vistos até agora referem-se a parâmetros por valor, ou seja, se alterarmos o valor do parâmetro **formal**, não afetaremos o parâmetro **real**.

Há duas formas de definirmos o posicionamento das sub-rotinas: No início (como visto até agora) e no final.

```
#include <stdio.h>
#include <stdlib.h>
float a,b,x,y;

float media(float n1, float n2){
    return (n1+n2)/2;
}

int main(){
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media(a,b));
    system("pause");
    return(0);
}
```

```
#include <stdio.h>
#include <stdlib.h>
float a,b,x,y;

float media(float n1, float n2);

int main(){
    printf("Entre com o primeiro valor: ");
    scanf("%f",&a);
    printf("Entre com o segundo valor: ");
    scanf("%f",&b);
    printf("media = %.2f\n",media(a,b));
    system("pause");
    return(0);
}

float media(float n1, float n2){
    return (n1+n2)/2;
}
```

Protótipo

Passagem de parâmetros

Todos os exemplos anteriores utilizamos PASSAGEM DE PARÂMETROS por **VALOR**, ou seja, sem qualquer possibilidade de modificar os valores dos parâmetros **reais** após a sua execução.

A passagem de parâmetros por **REFERÊNCIA** é caracterizada com um * (asterisco) antecedendo o valor do parâmetro **formal** no momento da definição da estrutura e utilizando um & na passagem do parâmetro **real**.


```
#include <stdio.h>
#include <stdlib.h>
```

```
void soma(int a, int *b){
    printf("soma = %d\n", a + *b);
    a=0;
    *b=0;
}
```

Perceba que há um ***** antecedendo o parâmetro formal **b**

```
int main(){
    int x,y;
    printf("Entre com um numero: ");
    scanf("%d",&x);
    printf("Entre com outro numero: ");
    scanf("%d",&y);
    soma(x, &y);
    printf("Primeiro valor informado: %d\n",x);
    printf(" Segundo valor informado: %d\n",y);
    printf("\n\n");
    system("PAUSE");
    return(0);
}
```

Perceba que há um **&** antecedendo o parâmetro real **b**

```
#include <stdio.h>
#include <stdlib.h>

void soma(int a, int *b) {
    printf("soma = %d\n", a + *b);
    a=0;
    *b=0;
}

int main() {
    int x,y;
    printf("Entre com um numero: ");
    scanf("%d",&x);
    printf("Entre com outro numero: ");
    scanf("%d",&y);
    soma(x, &y);
    printf("Primeiro valor informado: %d\n",x);
    printf(" Segundo valor informado: %d\n",y);
    printf("\n\n");
    system("PAUSE");
    return (0);
}
```

G:\CESUMAR_2022\Algoritmos II\aula 30-03-22\teste.exe

```
Entre com um numero: 10
Entre com outro numero: 35
soma = 45
Primeiro valor informado: 10
Segundo valor informado: 0

Pressione qualquer tecla para continuar. . .
```

```
#include <stdio.h>
#include <stdlib.h>

void soma(int a, int *b) {
    printf("soma = %d\n", a + *b);
    a=0;
    *b=0;
}

int main() {
    int x,y;
    printf("Entre com um numero: ");
    scanf("%d",&x);
    printf("Entre com outro numero: ");
    scanf("%d",&y);
    soma(x, &y);
    printf("Primeiro valor informado: %d\n",x);
    printf(" Segundo valor informado: %d\n",y);
    printf("\n\n");
    system("PAUSE");
    return (0);
}
```

G:\CESUMAR_2022\Algoritmos II\aula 30-03-22\teste.exe

```
Entre com um numero: 10
Entre com outro numero: 35
soma = 45
Primeiro valor informado: 10
Segundo valor informado: 0

Pressione qualquer tecla para continuar. . .
```

```
#include <stdio.h>
#include <stdlib.h>

void soma(int a, int *b) {
    printf("soma = %d\n", a + *b);
    a=0;
    *b=0;
}

int main() {
    int x,y;
    printf("Entre com um numero: ");
    scanf("%d",&x);
    printf("Entre com outro numero: ");
    scanf("%d",&y);
    soma(x, &y);
    printf("Primeiro valor informado: %d\n",x);
    printf(" Segundo valor informado: %d\n",y);
    printf("\n\n");
    system("PAUSE");
    return (0);
}
```

G:\CESUMAR_2022\Algoritmos II\aula 30-03-22\teste.exe

```
Entre com um numero: 10
Entre com outro numero: 35
soma = 45
Primeiro valor informado: 10
Segundo valor informado: 0

Pressione qualquer tecla para continuar. . .
```

```
#include <stdio.h>
#include <stdlib.h>

void soma(int a, int *b) {
    printf("soma = %d\n", a + *b);
    a=0;
    *b=0;
}

int main() {
    int x,y;
    printf("Entre com um numero: ");
    scanf("%d",&x);
    printf("Entre com outro numero: ");
    scanf("%d",&y);
    soma(x, &y);
    printf("Primeiro valor informado: %d\n",x);
    printf(" Segundo valor informado: %d\n",y);
    printf("\n\n");
    system("PAUSE");
    return (0);
}
```

G:\CESUMAR_2022\Algoritmos II\aula 30-03-22\teste.exe

```
Entre com um numero: 10
Entre com outro numero: 35
soma = 45
Primeiro valor informado: 10
Segundo valor informado: 0

Pressione qualquer tecla para continuar. . .
```

Valor de x

```
#include <stdio.h>
#include <stdlib.h>

void soma(int a, int *b) {
    printf("soma = %d\n", a + *b);
    a=0;
    *b=0;
}

int main() {
    int x,y;
    printf("Entre com um numero: ");
    scanf("%d",&x);
    printf("Entre com outro numero: ");
    scanf("%d",&y);
    soma(x, &y);
    printf("Primeiro valor informado: %d\n",x);
    printf(" Segundo valor informado: %d\n",y);
    printf("\n\n");
    system("PAUSE");
    return (0);
}
```

G:\CESUMAR_2022\Algoritmos II\aula 30-03-22\teste.exe

```
Entre com um numero: 10
Entre com outro numero: 35
soma = 45
Primeiro valor informado: 10
Segundo valor informado: 0
Pressione qualquer tecla para continuar. . .
```

Valor de y

Utilizamos passagem de parâmetro por **referência quando necessitamos modificar o valor do parâmetro real.**

Como exemplo, temos a TROCA efetuada ao classificarmos um vetor. Poderíamos criar um procedimento ao qual passaríamos os dois valores por parâmetro e o procedimento trocaria o valor dos parâmetros.

```
#include <stdio.h>
#include <stdlib.h>
void troca(int *a, int *b) {
    int tmp;
    tmp=*a;
    *a=*b;
    *b=tmp;
}
int main() {
    int x,y;
    printf("Entre com um numero: ");
    scanf("%d",&x);
    printf("Entre com outro numero: ");
    scanf("%d",&y);
    troca(&x, &y);
    printf("Primeiro valor informado: %d\n",x);
    printf(" Segundo valor informado: %d\n",y);
    printf("\n\n");
    system("PAUSE");
    return(0);
}
```

O exemplo ao lado troca o valor das variáveis **x** e **y** após executar o procedimento **troca**


```
#include <stdlib.h>
```

```
#include <time.h>
```

```
int x,y,i,vet[10];
```

Protótipos

```
void troca(int *a, int *b);
```

```
void sorteia();
```

```
void classifica();
```

```
//-----
```

```
int main(){
```

```
    srand(time(NULL));
```

```
    sorteia();
```

```
    classifica();
```

```
    system("PAUSE");
```

```
    return(0);
```

```
}
```

```
//-----
```

```
void sorteia(){
```

```
    for (i=0; i<10; i++)
```

```
        vet[i]=rand()%1000;
```

```
}
```

```
//-----
```

```
void troca(int *a, int *b){
```

```
    int tmp;
```

```
    tmp=*a;
```

```
    *a=*b;
```

```
    *b=tmp;
```

```
}
```

```
//-----
```

```
void classifica(){
```

```
    for (x=0; x<9; x++)
```

```
        for (y=x+1; y<10; y++)
```

```
            if (vet[y]<vet[x])
```

```
                troca(&vet[x], &vet[y]);
```

```
}
```

Programa principal

Procedimentos

Este outro exemplo, possui três procedimentos como protótipo.

É preenchido um vetor randomicamente e classificado em seguida. Note que dentro do procedimento **classifica** é executado o procedimento **troca**.

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int x,y,i,vet[10];

void troca(int *a, int *b);
void sorteia();
void classifica();
//-----

int main() {
    srand(time(NULL));
    sorteia();
    classifica();
    system("PAUSE");
    return(0);
}
//-----
```

```
void sorteia() {
    for (i=0; i<10; i++)
        vet[i]=rand()%1000;
}
//-----

void troca(int *a, int *b) {
    int tmp;
    tmp=*a;
    *a=*b;
    *b=tmp;
}
//-----

void classifica() {
    for (x=0; x<9; x++)
        for (y=x+1; y<10; y++)
            if (vet[y]<vet[x])
                troca(&vet[x], &vet[y]);
}
```

Para efetuar passagem de parâmetro com string, não haverá necessidade de utilizar o & antecedendo o parâmetro real ao chamar a função ou sub-rotina.

Exemplo:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
char nome1[50]="Ana";
char nome2[50]="Jose";
void mostra(char *s1, char*s2);
void troca(char *s1, char *s2);
//-----
int main() {
    mostra(nome1,nome2);
    troca(nome1,nome2);
    mostra(nome1,nome2);
    printf("\n\n\n");
    system("pause");
    return 0;
}
```

```
//-----
void mostra(char *s1, char*s2) {
    printf("%s \n",s1);
    printf("%s \n",s2);
    system("pause");
}
//-----
void troca(char *s1, char *s2) {
    char tmp[20];
    strcpy(tmp,s1);
    strcpy(s1,s2);
    strcpy(s2,tmp);
}
//
```

Exercícios (com sub-rotinas)

- 1) Escreva um programa que utilize uma sub-rotina para calcular e apresentar a série de Fibonacci de N termos. A série de *Fibonacci* é formada pela seqüência: 1, 1, 2, 3, 5, 8, 13, 21, 34, ... etc. Esta série caracteriza-se pela soma de um termo posterior com o seu anterior subsequente. Apresentar o resultado.
- 2) Criar um algoritmo que efetue o cálculo de uma prestação em atraso. Para tanto, utilize a fórmula $PREST = VALOR + (VALOR * (TAXA/100) * TEMPO)$. Apresentar o valor da prestação.
- 3) Desenvolva um algoritmo de programa que crie uma sub-rotina para calcular e apresentar o valor de uma potência de um número qualquer. Ou seja, ao informar para a sub-rotina o número e sua potência, deverá ser apresentado o seu resultado. Por exemplo, se for mencionado no programa principal a sub-rotina POTENCIA(2,3), deverá ser apresentado o valor 8.

Exercícios (com sub-rotinas)

- 4) Elaborar um programa que efetue a leitura de três valores (A, B e C) e apresente como resultado final o quadrado da soma dos três valores lidos.
- 5) Elaborar um programa que por meio de sub-rotina efetue a apresentação do valor da conversão em real (R\$) de um valor lido em dólar (US\$). Deverá ser solicitado por meio do programa principal o valor da cotação do dólar e a quantidade de dólar disponível.
- 6) Elaborar um programa que com o uso de sub-rotina do tipo função apresente o valor de uma temperatura em graus Celsius. O programa deverá ler a temperatura em graus Fahrenheit e apresentá-la convertida em graus Celsius. Sendo que a fórmula de conversão é $C \leftarrow (F - 32) * (5/9)$.

Exercícios (com sub-rotinas)

- 7) Elaborar um programa que possua uma sub-rotina que efetue e permita apresentar o somatório dos N primeiros números inteiros, definidos por um operador $(1+2+3+4+\dots+N)$.
- 8) Elaborar um programa que efetue a leitura de três valores (A, B e C) e apresente como resultado final a soma dos quadrados dos três valores lidos.