

MINIAULA DE ALGORITMOS FUNÇÕES

Prof. Ivanilton Polato

Departamento Acadêmico de Computação (DACOM-CM)

ipolato@utfpr.edu.br

1. Sem parâmetros e sem retorno

- São as funções mais simples, pois:
 - *Não recebem valores por parâmetros*
 - *Não devolvem valor como retorno*
 - *Apenas executam um trecho de código*
- Essas funções são do tipo VOID:
 - *Desobriga o retorno de valores*

Exemplo: soma de dois números

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

Declaração de bibliotecas

```
void soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

Função soma: solicita dois números, recebe os dados digitados pelo usuário, armazena em variáveis locais, faz a soma e apresenta o resultado em tela

```
void main() {  
    soma();  
    soma();  
    soma();  
}
```

Função principal (main), inicia a execução do programa e faz a chamada das funções para que a soma seja realizada

Exemplo!

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

```
void soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

```
void main() {  
    soma();  
    soma();  
    soma();  
}
```

A função **main()** inicia sua execução quando o programa tem início...

Exemplo!

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

```
void soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

```
void main() {  
    soma();  
    soma();  
    soma();  
}
```

A função **soma()** é chamada! Nesse ponto a função **main()** suspende sua execução e começa a execução da função **soma()** ...

Exemplo!

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

```
void soma() {
```

← Começa a execução da função **soma()** ...

```
    int n1, n2, s;
```

```
    printf("Primeiro número:\n");
```

```
    scanf("%d", &n1);
```

```
    printf("Segundo número:\n");
```

```
    scanf("%d", &n2);
```

```
    s = n1 + n2;
```

```
    printf("A soma é: %d\n", s);
```

```
}
```

```
void main() {
```

```
    soma() ;
```

```
    soma() ;
```

```
    soma() ;
```


```
}
```

Exemplo!

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

```
void soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

A função segue a
execução, linha a
linha, até o fim



```
void main() {  
    soma();  
    soma();  
    soma();  
}
```

Exemplo!

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

```
void soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

← A execução da função **soma()** termina e controle volta para a função **main()**

```
void main() {  
    soma();  
    soma();  
    soma();  
}
```


Exemplo!

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

```
void soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

```
void main() {  
    soma();  
    soma();  
    soma();  
}
```

Novamente a função **soma()** é chamada! E procede da mesma forma: a função **main()** suspende sua execução e começa a execução da função **soma()** ...

Exemplo!

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

```
void soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

```
void main() {  
    soma();  
    soma();  
    soma();  
}
```

- E assim o programa segue sua execução até o fim
- Sempre que uma função é chamada, a função em execução é suspensa para que a função que foi chamada possa ser executada
- Ao terminar a execução, a função que foi chamada é terminada (seus dados são destruídos) e o controle retorna para quem a chamou

2. Com parâmetros e sem retorno

- São as funções que:
 - *Recebem valores para processar*
 - *Não devolvem valor como retorno*
- Essas funções ainda são do tipo VOID:
 - *Desobriga o retorno de valores*

Exemplo: soma de dois números

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

```
void soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

```
void main() {  
    int x, y;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    soma(x, y);  
    soma(x, 10);  
    soma(10, 15);  
}
```

Função soma: RECEBE dois números por parâmetro, armazena em variáveis locais (parâmetros), faz a soma e apresenta o resultado na tela

Função principal (main), inicia a execução do programa e solicita dois números para o usuário, faz a chamada das funções passando os números via parâmetro para que a soma seja realizada dentro da função

Declaração de parâmetros

- Exemplo: `void soma(int n1, int n2) { ... }`
 - *Os parâmetros ficam entre os parênteses*
 - *Quantos forem necessários*
 - *Separados por vírgula*
 - *Cada qual com seu tipo de dados: char, int, long, float, double, dentre outros...*
- Quando declarados, obrigam a chamada da função a passar uma quantidade de parâmetros iguais à definida, na mesma ordem, e com os mesmo tipos (ou equivalentes)!

Exemplo!

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

```
void soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

```
void main() {
```

← A função **main()** inicia sua execução quando o programa tem início...

```
    int x, y;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    soma(x, y);  
    soma(x, 10);  
    soma(10, 15);
```

```
}
```

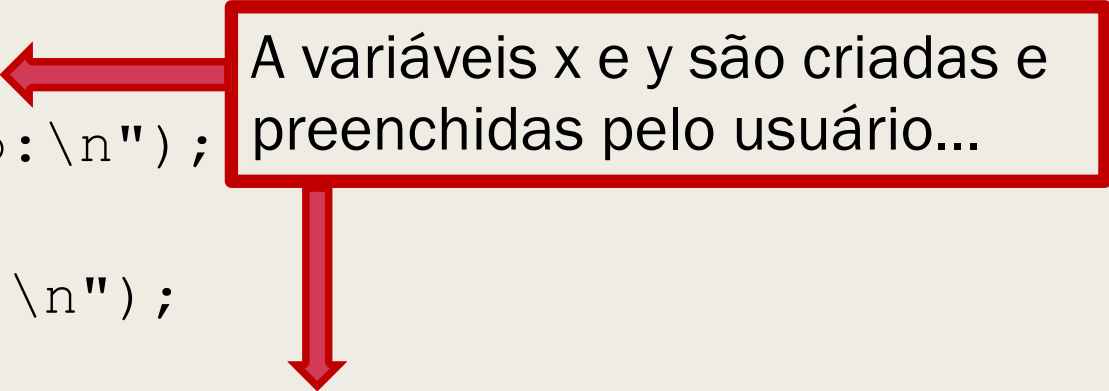
Exemplo!

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

```
void soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

```
void main() {  
    int x, y;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    soma(x, y);  
    soma(x, 10);  
    soma(10, 15);  
}
```

A variáveis x e y são criadas e preenchidas pelo usuário...



Exemplo!

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

```
void soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

```
void main() {  
    int x, y;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    soma(x, y);  
    soma(x, 10);  
    soma(10, 15);  
}
```

A função **soma()** é chamada! Nessa chamada devem, obrigatoriamente, ser passados os parâmetros na ordem esperada, em número igual e em tipos compatíveis: dois números inteiros!

A função **main()** suspende sua execução e começa a execução da função **soma()** ...

Nesse momento, os valores de x e y, SÃO COPIADOS, para n1 e n2 (da função soma), respectivamente...

Exemplo!

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

```
void soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

```
void main() {  
    int x, y;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    soma(x, y);  
    soma(x, 10);  
    soma(10, 15);  
}
```

Quando **soma()** inicia, **n1** e **n2**, recebem automaticamente, o valores que foram enviados pela chamada, ou seja, os valores de **x** e **y**, respectivamente.


As variáveis **n1** e **n2** são definidas nos parâmetros e não precisam ser definidas em outro lugar. São consideradas variáveis locais normais, e que sempre são inicializadas durante a chamada da função!

Exemplo!

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

```
void soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

A função segue a
execução, linha a
linha, até o fim



```
void main() {  
    int x, y;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    soma(x, y);  
    soma(x, 10);  
    soma(10, 15);  
}
```

Exemplo!

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

```
void soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    printf("A soma é: %d\n", s);  
}
```

← A execução da função **soma()** termina e controle volta para a função **main()**

```
void main() {  
    int x, y;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    soma(x, y);  
    soma(x, 10);  
    soma(10, 15);  
}
```

Exemplo: soma de dois números

- Os parâmetros da chamada podem variar conforme a necessidade! Observe:
- Duas variáveis inteiras:

soma (x, y) ;

- Uma variável inteira e um número inteiro:

soma (x, 10) ;

- Dois números inteiros:

soma (10, 15) ;



O importante é a ordem, a quantidade e o tipo!

3. Sem parâmetros e com retorno

- São as funções que:
 - *Não recebem valores por parâmetros*
 - *Retornam obrigatoriamente um valor*
- Essas funções utilizam tipos diferentes de VOID:
 - *char, int, long, float, double, dentre outros....*
- Devem usar a instrução **return** ao final da função!

Exemplo: soma de dois números

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

Declaração de bibliotecas

```
int soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    return s;  
}
```

Função soma: solicita dois números, recebe e armazena em variáveis locais, faz a soma e retorna o valor para a função chamadora! Observe que o tipo da função é **int**!

```
void main() {  
    int res;  
    res = soma();  
    printf("A soma é: %d\n", res);  
}
```

Função principal (main), inicia a execução do programa e faz a chamada da função soma, aguardando o retorno para que seja armazenado na variável res. Em seguida o valor é impresso em tela...

Exemplo!

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

```
int soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int res;  
    res = soma();  
    printf("A soma é: %d\n", res);  
}
```

A função **main()** inicia sua execução quando o programa tem início...

Exemplo!

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

```
int soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int res;  
    res = soma();  
    printf("A soma é: %d\n", res);  
}
```

A função **soma()** é chamada! A função **main()** suspende sua execução e fica aguardando o resultado da função **soma()** ...

Exemplo!

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

```
int soma () {
```



Começa a execução da função **soma ()** ...

```
    int n1, n2, s;
```

```
    printf("Primeiro número:\n");
```

```
    scanf("%d", &n1);
```

```
    printf("Segundo número:\n");
```

```
    scanf("%d", &n2);
```

```
    s = n1 + n2;
```

```
    return s;
```

```
}
```

```
void main() {
```

```
    int res;
```

```
    res = soma();
```

```
    printf("A soma é: %d\n", res);
```


```
}
```

Exemplo!

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

```
int soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    return s;  
}
```

A função segue a
execução, linha a
linha, até o fim



```
void main() {  
    int res;  
    res = soma();  
    printf("A soma é: %d\n", res);  
}
```

Exemplo!

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

```
int soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    return s;  
}
```

Ao final a instrução **return** é executada. É a última instrução da função antes de seu término e exclusão. Nesse momento, o valor contido na variável da instrução **return** (nesse caso, **s**) é COPIADO para a variável receptora na função chamadora. A função **soma ()** termina e o controle volta para a função **main ()** ...

```
void main() {  
    int res;  
    res = soma();  
    printf("A soma é: %d\n", res);  
}
```

Exemplo!

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

```
int soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int res;  
    res = soma();  
    printf("A soma é: %d\n", res);  
}
```

Quando a função **soma()** termina sua execução o valor retornado é armazenado na variável **res**, que estava esperando o retorno!

Exemplo!

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

```
int soma() {  
    int n1, n2, s;  
    printf("Primeiro número:\n");  
    scanf("%d", &n1);  
    printf("Segundo número:\n");  
    scanf("%d", &n2);  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int res;  
    res = soma();  
    printf("A soma é: %d\n", res);  
}
```

A função **main()** então termina sua execução com a impressão do valor da soma em tela!

Pontos importantes!

- Nenhuma instrução deve ser colocada após o **return**!
- A variável que vai receber o retorno da função deve ser do mesmo tipo da função declarada!
- A variável de retorno também deve ser do mesmo tipo da função declarada!

```
int soma() {  
    int ... s;  
    ...  
    return s;  
}
```

```
void main() {  
    int res;  
    res = soma();  
    printf("A soma é: %d\n", res);  
}
```

4. Com parâmetros e com retorno

- São funções mais elaboradas que:
 - *Recebem valores por parâmetros*
 - *Retornam obrigatoriamente um valor*
- Essas funções utilizam tipos diferentes de VOID:
 - *char, int, long, float, double, dentre outros....*
- Devem usar a instrução **return** ao final da função!

Exemplo: soma de dois números

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

```
int soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int x, y, res;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    res = soma(x, y);  
    printf("A soma é: %d\n", res);  
}
```

Função soma: RECEBE dois números, armazena em variáveis locais (parâmetros), faz a soma e retorna o resultado para a função chamadora!

Função principal (main), inicia a execução do programa e solicita dois números para o usuário, faz a chamada das funções passando os números via parâmetro para que a soma seja realizada. Recebe o retorno da função na variável res, e ao final imprime em tela o valor da soma!

Exemplo: soma de dois números

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

```
int soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int x, y, res;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    res = soma(x, y);  
    printf("A soma é: %d\n", res);  
}
```

A função **main()** inicia sua execução quando o programa tem início...

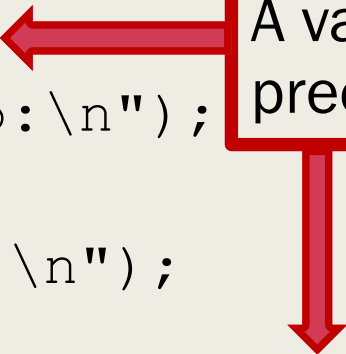
Exemplo: soma de dois números

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

```
int soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int x, y, res;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    res = soma(x, y);  
    printf("A soma é: %d\n", res);  
}
```

A variáveis são declaradas, e x e y preenchidas pelo usuário...



Exemplo: soma de dois números

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

```
int soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int x, y, res;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    res = soma(x, y);  
    printf("A soma é: %d\n", res);  
}
```

A função **soma()** é chamada! A função **main()** suspende sua execução e fica aguardando o retorno da função **soma()**. Nesse momento, os valores de **x** e **y**, SÃO COPIADOS, para **n1** e **n2** (da função **soma**), respectivamente...

Exemplo: soma de dois números

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

```
int soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    return s;  
}
```

Quando **soma()** inicia, n1 e n2, recebem automaticamente, o valores que foram enviados pela chamada, ou seja, os valores de x e y.

As variáveis n1 e n2 são definidas nos parâmetros e sempre são inicializadas durante a chamada da função!


```
void main() {  
    int x, y, res;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    res = soma(x, y);  
    printf("A soma é: %d\n", res);  
}
```

Exemplo: soma de dois números

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

```
int soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    return s;  
}
```

A função segue a
execução, linha a
linha, até o fim



```
void main() {  
    int x, y, res;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    res = soma(x, y);  
    printf("A soma é: %d\n", res);  
}
```

Exemplo: soma de dois números

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

```
int soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    return s;  
}
```

Ao final a instrução **return** é executada. Nesse momento, o valor contido na variável da `s` é COPIADO para a variável `res` na função chamadora. A função **soma()** termina e o controle volta para a função **main()** ...

```
void main() {  
    int x, y, res;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    res = soma(x, y);  
    printf("A soma é: %d\n", res);  
}
```

Exemplo: soma de dois números

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

```
int soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int x, y, res;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    res = soma(x, y);  
    printf("A soma é: %d\n", res);  
}
```

Quando a função **soma ()** termina sua execução o valor retornado é armazenado na variável **res**, que estava esperando o retorno!

Exemplo: soma de dois números

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

```
int soma(int n1, int n2) {  
    int s;  
    s = n1 + n2;  
    return s;  
}
```

```
void main() {  
    int x, y, res;  
    printf("Primeiro número:\n");  
    scanf("%d", &x);  
    printf("Segundo número:\n");  
    scanf("%d", &y);  
    res = soma(x, y);  
    printf("A soma é: %d\n", res);  
}
```

A função **main()** então termina sua execução com a impressão do valor da soma em tela!

Atenção!

- Sempre conferir:
 - *Ordem, quantidade e tipos dos parâmetros*
- Na declaração da função e na chamada da função!
- Sempre declarar uma função antes de seu uso, isto é, em geral, antes da função main
 - *Exceção: uso de protótipos de função*

Protótipos de função

- Em qualquer programa, podemos escrever funções antes ou depois da função main.
- Se optarmos por escrevê-las antes, nenhum cuidado especial será necessário.
- Se optarmos por escrevê-las abaixo da função main, devemos fazer uso dos protótipos de função.
 - *É uma declaração do cabeçalho da função seguida de ponto e vírgula (;)*



O uso de protótipos de função pode auxiliar quando o arquivo contém muitas funções interdependentes!

Exemplo de protótipo!

```
#include <stdio.h>

int soma(int n1, int n2); ←

void main() {
    int x, y, res;
    printf("Primeiro número:\n");
    scanf("%d", &x);
    printf("Segundo número:\n");
    scanf("%d", &y);
    res = soma(x, y);
    printf("A soma é: %d\n", res);
}

int soma(int n1, int n2) { ←
    int s;
    s = n1 + n2;
    return s;
}
```

O uso do protótipo desobriga a declaração da função antes de ser utilizada pelo main!

Nesse caso, a função pode ser desenvolvida em qualquer ponto do seu arquivo de código fonte!