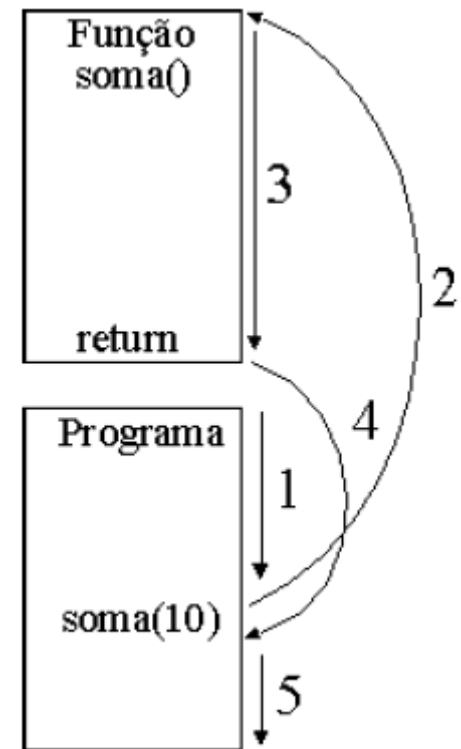


Funções na linguagem C

- módulos que agrupam um conjunto de comandos que são executados quando a função é chamada
- executam uma tarefa específica
- retornam um único valor (ou vazio = void)
- devem ser declaradas antes do uso (chamada)
- após sua execução, o fluxo do programa retorna ao ponto imediatamente após a chamada da função



Funções: sintaxe

```
<tipo_ret> nome_função (lista_de_parâmetros) {  
    // declarações locais  
    // corpo da função  
}
```

- **tipo_ret**: tipo de valor retornado pela função (default: inteiro)
- **nome_função**: identificador da função
- **lista_de_parâmetros**: tipo de cada parâmetro seguido do identificador, separados por vírgulas (parâmetros formais)
 - sem parâmetros → parênteses obrigatórios

Prototype de uma função → “declaração” da função

Função **SEM** parâmetros

sem retorno

```
void media(){
    int a, b;
    float m;

    printf("digite 2 valores\n");
    scanf("%d %d", &a, &b);
    m = (float)(a+b)/2;
    printf("media = %.2f\n", m);
}
```

```
int main(){
    media();
    return 0;
}
```

Função **SEM** parâmetros

sem retorno

```
void media(){
    int a, b;
    float m;

    printf("digite 2 valores\n");
    scanf("%d %d", &a, &b);
    m = (float)(a+b)/2;
    printf("media = %.2f\n", m);
}
```

```
int main(){
    media();
    return 0;
}
```

com retorno

```
float media(){
    int a, b;
    float r;

    printf("digite 2 valores\n");
    scanf("%d %d", &a, &b);
    r = (float)(a+b)/2;
    return r;
}
```

```
int main(){
    float m;

    m = media();
    printf("media = %.2f\n", m);
    return 0;
}
```

Parâmetros

- forma de comunicação entre funções
 - declarados no cabeçalho das funções (parâmetros formais) - são locais as funções
 - uma função com parâmetros deve ser chamada com argumentos de igual número e tipo
 - Passagem de parâmetros
 - define sobre quais dados a função deve operar
 - **Passagem de parâmetros por valor**
 - Passagem de parâmetros por referência
-

Passagem de parâmetros por valor

- Passa uma cópia do valor
- Alterações no valor dos parâmetros não afetam o conteúdo da variável no escopo da função principal
- O parâmetro se comporta como uma variável local

Função **COM** parâmetros (passagem por valor)

sem retorno

```
void media(int x, int y){  
    float m;  
    m = (float)(x+y)/2;  
    printf("media = %.2f\n", m);  
}
```

```
int main(){  
    int a, b;  
    printf("digite 2 valores\n");  
    scanf("%d %d", &a, &b);  
    media(a, b);  
    return 0;  
}
```

Função **COM** parâmetros (passagem por valor)

sem retorno

```
void media(int x, int y){  
    float m;  
    m = (float)(x+y)/2;  
    printf("media = %.2f\n", m);  
}
```

```
int main(){  
    int a, b;  
    printf("digite 2 valores\n");  
    scanf("%d %d", &a, &b);  
    media(a, b);  
    return 0;  
}
```

com retorno

```
float media(int x, int y){  
    float r;  
    r = (float)(x+y)/2;  
    return r;  
}
```

```
int main(){  
    int a, b;  
    float m;  
  
    printf("digite 2 valores\n");  
    scanf("%d %d", &a, &b);  
    m = media(a, b);  
    printf("media = %.2f\n", m);  
    return 0;  
}
```