

INSTITUTO FEDERAL

Rio Grande do Sul

Campus Porto Alegre

Linguagem de Programação I

Prof. Fabio Okuyama

Curso Superior em Tecnologia em Sistemas para Internet

Funções

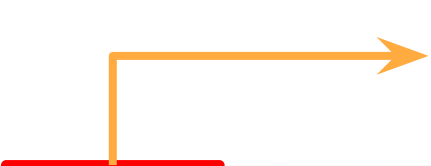
subprogramas

Funções – o que é?

- Conjunto de comandos que **realiza uma tarefa específica.**
- Bloco de código de programa;
- Pode ser utilizado várias vezes;
- Melhorar legibilidade do código;
- Estrutura do código;
- Possibilitar **REUSO**;
- Pode conter: Valores de Entrada e um valor de Saída (Retorno)

Funções - estrutura

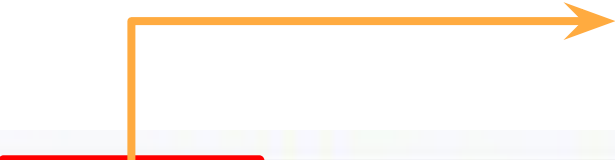
Tipo de retorno



```
1 int media (int a, int b){  
2     printf ("Calculando...");  
3     return((a+b)/2);  
4 }
```

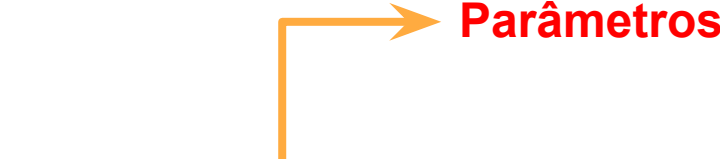
Funções - estrutura

**Nome da
função**



```
1 int media (int a, int b){  
2     printf ("Calculando...");  
3     return((a+b)/2);  
4 }
```

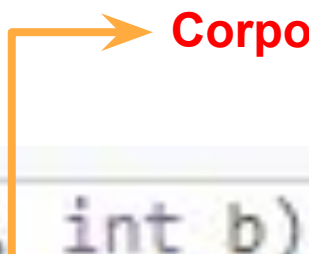
Funções - estrutura



Parâmetros

```
1 int media (int a, int b){  
2     printf ("Calculando...");  
3     return((a+b)/2);  
4 }
```

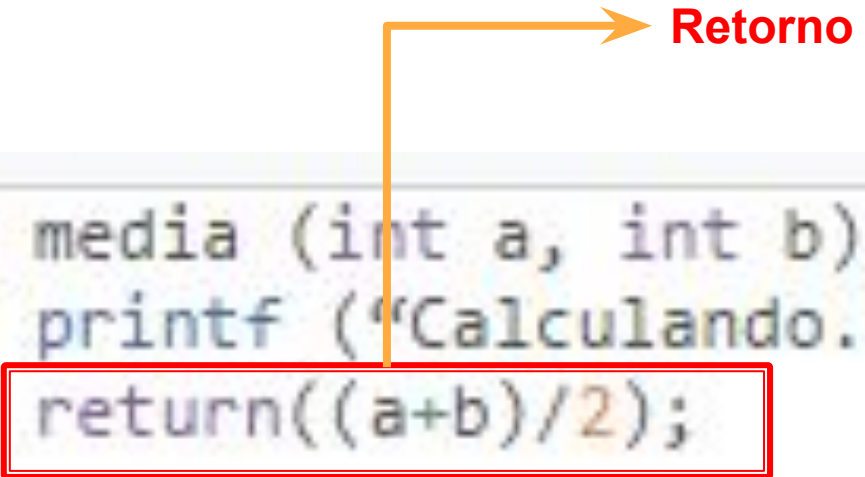
Funções - estrutura



Corpo

```
1 int media (int a, int b){  
2     printf ("Calculando...");  
3     return((a+b)/2);  
4 }
```

Funções - estrutura



```
1 int media (int a, int b){  
2     printf ("Calculando...");  
3     return((a+b)/2);  
4 }  
5
```

Retorno

Funções - chamadas

```
1  #include <stdio.h>
2
3  int media(int a, int b){
4      printf("\n...Calculando...\n");
5      return (a+b)/2;
6  }
7
8  int main(){
9      int resposta = media (10,20);
10     printf("a media de 10 e 20 é : %d\n", resposta);
11
12     printf("a media de 30 e 50 é: %d", media(30,50));
13     return 0;
14 }
```

...Calculando...

a media de 10 e 20 é : 15

...Calculando...

a media de 30 e 50 é: 40

Nomes de funções

como nomear suas funções

Regras para nomes de funções

- **Mesma das variáveis**

- Deve começar com letra ou _ (Underscore)
- O nome de uma Função/variável não pode ser igual a uma palavra reservada (palavra-chave);
- nem igual ao nome de outra função declarada pelo programador, ou pelas bibliotecas do C;
- Limite de 32 caracteres

Parâmetros de uma função

alimentando sua função

Parâmetros de uma função

- Passagem de parâmetros por valor
 - As variáveis são definidas em parênteses:
 - `int func(int a, int b);`
 - `char func(char a, int b);`
 - Os parâmetros podem ser acessados dentro da função
 - A mudança de valor de um parâmetro não altera o valor do original (é feita uma cópia)

Parâmetros: que parâmetros?

o que passar como parâmetro:

função comprarTV(Marca, Tamanho, modelo);

função procurarCasa (quartos, vagas garagem, banheiros, tamanho);

função lavarCarro (Carro, com Cera, hora entrega);

função fazerCamiseta(Tamanho, Cor, Estampa);

Tipos de retorno

o que sua função deve retornar?

Tipos de retorno de uma função

- As funções deverão ter um tipo de retorno
 - São válidos os mesmo tipos disponíveis para variáveis (INT, CHAR, FLOAT, ...)
 - Além disso é possível definir uma função como VOID (sem retorno).
 - A variável ou o valor a ser retornado é definido pelo comando RETURN:
 - `return (x);`
 - `return(0);`

Retorno: O que retornar?

Quando você pede para alguém fazer algo, você pode ter um retorno ou não.

Você pede para entregarem algo (uma pizza?)

Função queroPizza("Calabresa", "G", "Coronel Vicente, 281");

Possíveis retornos:

- A Pizza;
- Nome da pessoa que recebeu a pizza;
- Nome do entregador que está entregando a pizza;
- Cobrança
- "Nada"

Retorno: o que retornar?

Função que faz um cálculo matemático

- retornar o resultado

Função que imprime algo na tela

- nada (void)
- número de caracteres impressos
- informação sobre onde foram impressos

Função que faz a leitura de um caractere do teclado

- um caracter (é isso que a função `getchar()` faz)

Exemplos de funções

funções na prática

Cotacao de hoje: 1 dolar vale R\$5,50
R\$100,00 valem U\$18.18
U\$100,00 valem R\$550.00

Exemplo de função

```
1  #include <stdio.h>
2
3  float dolar (float valorReais, float cotacao){
4      return valorReais/cotacao;
5  }
6  float reais (float valorDolar, float cotacao){
7      return valorDolar*cotacao;
8  }
9
10 int main()
11 {
12     printf("Cotacao de hoje: 1 dolar vale R$5,50\n");
13     printf("R$100,00 valem U$%.2f\n",dolar(100,5.5));
14     printf("U$100,00 valem R$%.2f\n",reais(100,5.50));
15
16     return 0;
17 }
```

Exemplos

```
//função que recebe peso e altura  
float imc (float peso, float altura){  
    return peso/altura*altura;  
}//retorna IMC
```

```
void classificaIMC(float indice){  
    if(indice>30) printf("Obeso");  
    else if(indice>25) printf("Sobrepeso");  
    else if (indice>18.5) printf("Peso ideal");  
    else printf("Abaixo do Peso");  
}//escreve em tela a classificação, de acordo com imc
```

Exemplo

Escreva programa que Leia turno e língua. Esse programa deve ter uma função receba como parametros: Língua e Turno e imprima na tela a saudação correta. os parametros para lingua são: 1 portugues, 2 inglês. Para o turno o parâmetro é dado pelos caracteres 'm', 't' e 'n'. O programa deve repetir o processo até que a língua informada seja '0' (zero);

Casos de teste:

Língua	Turno	Resposta
Português (1)	manhã (m)	BOM DIA
Português (1)	tarde (t)	BOA TARDE
Português (1)	noite (n)	BOA NOITE
Inglês (2)	manhã (m)	GOOD MORNING
Inglês (2)	tarde (t)	GOOD AFTERNOON
Inglês (2)	noite (n)	GOOD NIGHT

Exemplo (continuação)

variáveis

```
char turno; // vai guardar o caractere que indica o turno
int lingua;  // vai guardar o inteiro que indica a lingua
```

Rascunho do que deve ser feito:

```
enquanto lingua é diferente de 0
    ler turno;
    ler língua;
    chamar função;
```

Saudação em Português, turno tarde: Bom tarde!

Saudação em Inglês, turno noite: Good night!

1 possibilidade de solução:

```
1 #include <stdio.h>
2
3 void saudacao(int l, char t){
4     if(l==1){
5         switch(t){
6             case 'm': printf("Bom dia!");break;
7             case 't': printf("Bom tarde!");break;
8             case 'n': printf("Bom norte!");break;
9         }
10    } else if(l==2){
11        switch(t){
12            case 'm': printf("Good Morning!");break;
13            case 't': printf("Good Afternoon!");break;
14            case 'n': printf("Good night!");break;
15        }
16    }
17 }
19 int main(){
20     printf("Saudação em Português, turno tarde: ");
21     saudacao(1,'t');
22
23     printf("\n\nSaudação em Inglês, turno noite: ");
24     saudacao(2,'n');
25     return 0;
26 }
```


Escopo de variáveis

além dos blocos

Escopo de variáveis

Globais

- Podem ser acessadas de qualquer ponto
- Declaradas no fora de funções
- Ou no arquivo de header (cabeçalho)

Locais

- Declarados em um subprograma
- Podem ser acessadas apenas dentro do subprograma

Encontre a(s) linha(s) que darão erro:

```
1  #include <stdio.h>
2  int a;
3  int mensagem (char msg[]){
4      int b;
5      printf ("Ola %s!", msg);
6      printf("%d", a);
7      printf("%d", b);
8      printf("%d", c);
9  }
10
11 int main(){
12     int c;
13     printf("%d", a);
14     printf("%d", b);
15     printf("%d", c);
16 }
```

Cuidado com variáveis Globais

- Podem ser acessada de qualquer ponto
- Risco de erros lógicos
 - Difíceis de corrigir

Assinatura de funções

a identidade de uma função

Assinaturas de funções

- Servem para informar ao compilador a existência de uma função que vai ser utilizada (chamar a função antes da implementação)
- Normalmente ficam no arquivo Header (.h) – Cabeçalho
- Os identificadores dos parâmetros podem ser omitidos

Exemplo de funções: assinatura

```
1 #include <stdio.h>
2
3 void saudacao(int, char);
4
5 int main() {
6     int lingua;
7     char turno;
8     do{
9         printf("\nDigite o turno (m/t/n): ");
10        scanf(" %c",&turno);
11        printf("Digite a Lingua (1:Pt / 2:En)");
12        scanf("%d",&lingua);
13        saudacao(lingua,turno);
14    }while(lingua!=0);
15 }
16
17 void saudacao(int l, char t){
18     if(l==1){
19         switch(t){
20             case 'm': printf("Bom dia!"); break;
21             case 't': printf("Boa tarde!"); break;
22             case 'n': printf("Boa noite!"); break;
23         }
24     }else if(l==2){
25         switch(t){
26             case 'm': printf("Good morning!"); break;
27             case 't': printf("Good afternoon!"); break;
28             case 'n': printf("Good night!"); break;
29         }
30     }
31 }
```

```
Digite o turno (m/t/n): m
Digite a Lingua (1:Pt / 2:En)1
Bom dia!
Digite o turno (m/t/n): t
Digite a Lingua (1:Pt / 2:En)2
Good afternoon!
Digite o turno (m/t/n): m
Digite a Lingua (1:Pt / 2:En)0
```

Mais exemplos...

```
1  #include <stdio.h>
2
3  void mult(){
4      float a,b;
5      printf("digite um valor: ");
6      scanf("%f",&a);
7      printf("digite outro valor: ");
8      scanf("%f",&b);
9      printf("%.3f", a*b);
10 }
11
12 int main(){
13     mult();
14 }
```

Mais exemplos...

```
1  #include <stdio.h>
2
3  float prod(float x, float y){
4      return x*y;
5  }
6
7  int main(){
8      float a,b;
9      printf("digite a base do retangulo: ");
10     scanf("%f",&a);
11     printf("digite a altura do retangulo: ");
12     scanf("%f",&b);
13     printf("area do retangulo: %.1f",prod(a,b));
14 }
```