



Introdução à Programação

Eduardo Silva Lira

XLVIII Programa de Verão do IME-USP

São Paulo - SP, Jan 2019



INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
UNIVERSIDADE DE SÃO PAULO



Funções: **cabeçalho**

Questão:

```
/*Exemplo de função para somar dois valores*/
#include<stdio.h>

int main(){
    int a = 5, b = 7;
    printf("soma: %d\n", somar(a, b));
    return 0;
}

int somar(int a, int b){
    int resultado;
    resultado = a + b;
    return resultado ;
}
```

Conseguiremos compilar e rodar este programa?

Funções: **cabeçalho**

- Definir a estrutura básica da função
 - Tipo de retorno
 - Identificador
 - Parâmetros
- Implementação posterior
- Aprimora o funcionamento do compilador

Funções: **cabeçalho**

Exemplo:

```
/*Exemplo de função para somar dois valores*/
#include<stdio.h>

int somar(int, int);

int main(){
    int a = 5, b = 7;
    printf("soma: %d\n", somar(a, b));
    return 0;
}

int somar(int a, int b){
    int resultado;
    resultado = a + b;
    return resultado ;
}
```

Funções: **cabeçalho**

Exemplo:

```
/*Exemplo de função para somar dois valores*/  
#include<stdio.h>
```

```
int somar(int, int);
```

```
int main(){  
    int a = 5, b = 7;  
    printf("soma: %d\n", somar(a, b));  
    return 0;  
}
```

```
int somar(int a, int b){  
    int resultado;  
    resultado = a + b;  
    return resultado ;  
}
```

Aqui nós
colocamos o
cabeçalho da
função

Aqui nós
implementamos
a função

Funções: **cabeçalho**

- Como se diz cabeçalho em inglês?
- O que quer dizer o “.h” no **#include<stdio.h>**?
- Onde isto irá nos levar?

Funções: **cabeçalho**

- Como se diz cabeçalho em inglês?
- O que quer dizer o “.h” no **#include<stdio.h>**?
- Onde isto irá nos levar?
 - Lembram do “*Escreva uma vez, utilize onde quiser*”?

Funções: **cabeçalho**

- Como se diz cabeçalho em inglês?
- O que quer dizer o “.h” no **#include<stdio.h>**?
- Onde isto irá nos levar?
 - Lembram do “*Escreva uma vez, utilize onde quiser*”?

Iremos criar nossas próprias **bibliotecas**

Como criar minhas bibliotecas?

Crie um arquivo de **header**, com os **protótipos** de funções. Salve como **minhalib.h**

```
#ifndef _MINHALIB_H_
#define _MINHALIB_H_

/*Cabecalho das minhas funcoes*/
int  somar(int, int);
double dobro(double);

#endif
```

Como criar minhas bibliotecas?

- Crie um arquivo **minhalib.c**
- Inclua o arquivo de **header**
 - **#include “minhalib.h”**
 - Implemente **todas** as funções
 - Programe!

Como criar minhas bibliotecas?

Arquivo **minhalib.c**:

```
#include "minhalib.h"

int somar(int a, int b) {
    return a + b;
}

double dobro(double x) {
    return x * 2;
}
```

Como criar minhas bibliotecas?

- Preparar a biblioteca para ser utilizada

- Geração de código objeto

**gcc -Wall -ansi -pedantic-errors -c
minhalib.c -o minhalib.o**

- Agora utilize a biblioteca em seus programas!

Como criar minhas bibliotecas?

Arquivo **expLib.c**: utilizar a biblioteca criada

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

int main() {
    int a = 10, b = 70;
    double x = 5.7, y = 7.0;
    printf("Soma de a e b: %d\n", somar(a, b));
    printf("Dobro de x: %f\n", dobro(x));
    printf("Dobro de y: %f\n", dobro(y));
    return 0;
}
```

Como criar minhas bibliotecas?

- Compile seu programa
 - Observe que a biblioteca vai no comando!

gcc -Wall -ansi -pedantic-errors

expLib.c -o expLib minhalib.o

Como criar minhas bibliotecas?

- Compile seu programa
 - Observe que a biblioteca vai no comando!

gcc -Wall -ansi -pedantic-errors

expLib.c -o expLib **minhalib.o**

Como criar minhas bibliotecas?

- Compile seu programa
 - Observe que a biblioteca vai no comando!

gcc -Wall -ansi -pedantic-errors

expLib.c -o expLib **minhalib.c**

Tente também adicionar direto o arquivo .c

Dicas! Bibliotecas e funções

- `stdio.h`
 - `remove`
 - `rename`
 - `fread`
 - `fwrite`

Dicas! Bibliotecas e funções

- ctype.h
 - isalpha(c)
 - isalphanum(c)
 - isupper(c)
 - islower(c)
 - iscntrl(c)
 - tolower(c)
 - toupper(c)

Dicas! Bibliotecas e funções

- string.h
 - char *strcpy(s, cp)
 - int strcmp(cs, cp)
 - size_t strlen(cs)
 - char *strncat(s,ct,n)

Dicas! Bibliotecas e funções

- `math.h`
 - `sin(x)`
 - `cos(x)`
 - `tan(x)`
 - `log10(x)`
 - `pow(x, y)`
 - `sqrt(x)`
 - `floor(x)`

Dicas! Bibliotecas e funções

- time.h
 - time_t time(time_t *tp)
 - double difftime(time_t time2, time_t time1)

Dicas! Bibliotecas e funções

- `stdlib.h`
 - `int atoi(const char *s)`
 - `long atol(const char *s)`
 - `int system(const char *s)`
 - `int abs(int n)`
 - `int rand(void)`
 - `void srand(unsigned int seed)`

Dicas! Bibliotecas e funções

- limits.h e float.h
 - Constantes com os maiores e menores valores armazenáveis
 - CHAR_MAX
 - LONG_MAX
 - SHRT_MIN
 - DBL_MIN

Dúvidas?