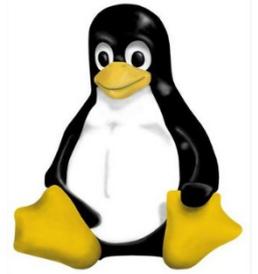


INSTITUTO FEDERAL  
BAIANO



# LP II

# Estrutura de Dados

---

## Funções e Procedimentos

*Prof. José Honorato F. Nunes*

**honorato.nunes@ifbaiano.bonfim.edu.br**

**honoratonunes@softwarelivre.org**

# Resumo da aula

---

- Sub-rotinas
  - Procedimentos
  - Funções

# Sub-rotina

---

- Sub-rotina é um programa que auxilia o programa principal através da realização de uma determinada subtarefa.
- Costuma receber os nomes de sub-programa, procedimento, método ou módulo.
- São chamados dentro do corpo do programa principal como se fossem comandos.
- Após seu término, a execução continua a partir do ponto onde foi chamado.
- A chamada de um subprograma simplesmente gera um desvio provisório no fluxo de execução.

---

# **SUB-ROTINAS → Procedimientos**

Prof. José Honorato F. Nunes  
honoratonunes@gmail.com

# Procedimentos

---

- Procedimento é uma sub-rotina que não retorna nenhum valor.
- Ao criarmos um procedimento em C, colocamos a palavra **void** antes do nome da sub-rotina, indicando a ausência de um valor de retorno.
- **main()** também é uma função. Sua única diferença é que consiste em ser a função automaticamente executada após o programa ser carregado.

# Procedimento: exemplo

---

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

```
void soma (int x, int y) {  
    printf("%d", x+y);  
}
```

```
int main () {  
    int x, y;  
    x=2; y=3;  
    soma(x,y);  
    return 0;  
}
```

# Atividade: programa com Procedimento

---

- Fazer um algoritmo para receber o valor inteiro não negativo  **$n$** , e usar um procedimento para calcular e imprimir o valor de seu fatorial.

---

# **SUB-ROTINAS → Função**

# Função

---

- Há um caso particular de sub-rotina que recebe o nome de função.
- Uma função, além de executar uma determinada tarefa, retorna um valor para quem a chamou, que é o resultado da sua execução.
- Por este motivo, a chamada de uma função aparece no corpo do programa principal como uma expressão, e não como um comando.

# Função: exemplo

---

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

```
int produto (int k, int z){  
    return (k * z);  
}
```

```
int main () {  
    int x, y, r;  
    x=2; y=3;  
    r = produto(x,y);  
    printf("%d", r);  
    return 0;  
}
```

# Função\Procedimento

---

- C exige que se coloque o protótipo da função e\ou procedimento antes dela ser chamada, se estas forem implementadas depois da função ***main()***.
- No protótipo da função e\ou procedimento não há necessidade de indicarmos os nomes dos parâmetros, apenas os seus tipos.

# Função x Procedimento - Exemplo

```
#include <stdio.h>
void soma (int,int);
int produto (int,int);
int main () {
    int x, y, r;
    x=2; y=3;
    soma(x,y);
    r = produto(x,y);
    printf("\nproduto = %d", r);
    return 0;
}
int produto (int k, int z){
    return (k * z);
}
void soma (int x, int y) {
    printf("soma= %d", x+y);
}
```