



UNIVERSIDADE  
FEDERAL DO CEARÁ



# Programação

## (CK0226 – 2022.2)

Curso: Ciência da Computação

Professor: Lincoln Souza Rocha

E-mail: [lincoln@dc.ufc.br](mailto:lincoln@dc.ufc.br)

# Introdução à Linguagem C – Parte II

# Sumário

- Controle de Fluxo
  - Decisão
  - Laço
  - Seleção

# Controle de Fluxo: Decisão

# Tomada de Decisão (Comando if)

```
if(<expressão>) {  
    <bloco de comandos if>  
} else {  
    <bloco de comandos else>  
}
```

OU

```
if(<expressão>) {  
    <bloco de comandos>  
}
```

**Nota!**

verdadeiro é  $\neq 0$  e  
falso é  $= 0$ .

# Tomada de Decisão (Comando if)

Se a temperatura for menor do que 20°C, então está frio

Se a temperatura estiver entre 20°C e 30°C, então está agradável

Se a temperatura for maior do que 30°C, então está quente

# Tomada de Decisão (Comando `if`)

```
#include <stdio.h>

int main (void) {
    int temp;
    printf("Digite a temperatura: ");
    scanf("%d", &temp);

    if (temp < 30)
        if (temp > 20)
            printf("Temperatura agradável. \n");
    else
        printf("Temperatura quente. \n");

    return 0;
}
```

# Tomada de Decisão (Comando `if`)

```
#include <stdio.h>

int main (void) {
    int temp;
    printf("Digite a temperatura: ");
    scanf("%d", &temp);

    if (temp < 30)
        if (temp > 20)
            printf("Temperatura agradável. \n");
    else
        printf("Temperatura quente. \n");

    return 0;
}
```



# Tomada de Decisão (Comando `if`)

```
#include <stdio.h>

int main (void) {
    int temp;
    printf("Digite a temperatura: ");
    scanf("%d", &temp);

    if (temp < 30) {
        if (temp > 20)
            printf("Temperatura agradável. \n");
    } else
        printf("Temperatura quente. \n");

    return 0;
}
```

# Tomada de Decisão (Comando `if`)

```
#include <stdio.h>

int main (void) {
    int temp;
    printf("Digite a temperatura: ");
    scanf("%d", &temp);

    if (temp < 10)
        printf("Temperatura muito fria\n");
    else if (temp < 20)
        printf("Temperatura fria\n");
    else if (temp < 30)
        printf("Temperatura agradável\n");
    else
        printf("Temperatura quente\n");

    return 0;
}
```

# Operador Ternário (Comando “?”)

**<condição> ? <expressão> : <expressão>;**

Exemplo: `maximo = (a > b) ? a : b;`

**Equivalente if:**

```
if (a > b)
    maximo = a;
else
    maximo = b;
```

# Controle de Fluxo: Laço

# Construção de Laços

Fatorial de um número inteiro não negativo:

$$n! = n \times (n-1) \times (n-2) \dots 3 \times 2 \times 1$$

$$\text{onde : } 0! = 1$$

# Construção de Laços

Definição recursiva da função fatorial:

- $\text{fatorial}(0) = 1$
- $\text{fatorial}(n) = n * \text{fatorial}(n-1)$

Definição não recursiva de  $\text{fatorial}(n)$ :

- Comece com:  $k = 1$  e  $f = 1$
- Faça enquanto  $k \leq n$ 
  - $f = f * k$
  - incremente  $k$

# Laços (Comando while)

```
while (<expressão>) {  
    <bloco de comandos>  
}
```

# Laços (Comando while)

```
#include <stdio.h>

int main (void) {
    int k, n;
    long int f = 1;
    printf("Digite um numero inteiro nao negativo:");
    scanf("%d", &n);

    k = 1;
    while (k <= n) {
        f = f * k;
        k = k + 1;
    }
    printf(" Fatorial = %d \n", f);
    return 0;
}
```



# Laços (Comando for)

```
for (<expressão>; <expressão>; <expressão>) {  
    <bloco de comandos>  
}
```

<expressão>: expressão inicial

<expressão>: expressão lógica

<expressão>: expressão de incremento

# Laços (Comando for)

```
#include <stdio.h>

int main (void) {
    int k, n;
    long int f = 1;
    printf("Digite um numero inteiro nao negativo:");
    scanf("%d", &n);

    for(k = 1; k <= n; k = k + 1) {
        f = f * k;
    }
    printf(" Fatorial = %d \n", f);
    return 0;
}
```

# Laços (Comando for)

```
#include <stdio.h>

int main (void) {
    int k, n;
    long int f = 1;
    printf("Digite um numero inteiro nao negativo:");
    scanf("%d", &n);

    for(k = 1; k <= n; k + 1) {
        f = f * k;
    }
    printf(" Fatorial = %d \n", f);
    return 0;
}
```

**O que acontece  
com o programa?**

# Laços (Comando do-while)

```
do {  
    <bloco de comandos>  
} while (<expressão>)
```

# Laços (Comando do-while)

```
#include <stdio.h>

int main (void) {
    int k, n;
    long int f = 1;
    do {
        printf("Digite um numero inteiro nao negativo:");
        scanf("%d", &n);
    } while(n < 0);

    for(k = 1; k <= n; k = k + 1) {
        f = f * k;
    }
    printf(" Fatorial = %d \n", f);
    return 0;
}
```

# Interrupção de Laços

- Comando `break`
  - Termina a execução do comando de laço
- Comando `continue`
  - Termina a iteração corrente e passa para a próxima

# Laços (Comando break)

```
#include <stdio.h>

int main (void) {
    int i;
    for(i = 0; i < 10; i++) {
        if(i == 5)
            break;
        printf(" %d ", i);
    }
    printf("fim\n");
    return 0;
}
```

# Laços (Comando continue)

```
#include <stdio.h>

int main (void) {
    int i;
    for(i = 0; i < 10; i++) {
        if(i == 5)
            continue;
        printf(" %d ", i);
    }
    printf("fim\n");
    return 0;
}
```



# Controle de Fluxo: Seleção

## Seleção (Comando `switch`)

```
switch (<expressão>) {  
    case op1: <bloco de comandos>; break;  
    case op2: <bloco de comandos>; break;  
    (...)  
    default: <bloco de comandos> break;  
}
```

# Seleção (Comando switch)

```
#include <stdio.h>
int main (void) {
    float num1, num2;
    char op;
    printf("Digite: numero op numero\n");
    scanf ("%f %c %f", &num1, &op, &num2);
    switch (op){
        case '+': printf(" = %f\n", num1+num2); break;
        case '-': printf(" = %f\n", num1-num2); break;
        case '*': printf(" = %f\n", num1*num2); break;
        case '/': printf(" = %f\n", num1/num2); break;
        default: printf("Operador invalido!\n"); break;
    }
    return 0;
}
```



UNIVERSIDADE  
FEDERAL DO CEARÁ



# Programação

## (CK0226 – 2022.2)

Curso: Ciência da Computação

Professor: Lincoln Souza Rocha

E-mail: [lincoln@dc.ufc.br](mailto:lincoln@dc.ufc.br)