

# Linguagem C

## (estruturas condicionais)

André Tavares da Silva

[andre.silva@udesc.br](mailto:andre.silva@udesc.br)

# Comandos de Controle de Fluxo

- Todos os comandos devem ser terminados com um '!'.
- '{' e '}' são usados para delimitar um bloco de comandos.
- Não é necessário colocar ponto-e-vírgula após '}'.

# if

```
if (condição)  
    comando;
```

```
if (condição) {  
    bloco de comandos;  
}
```

- Se a condição for verdadeira, o comando ou bloco de comandos que forma o corpo do *if* é executado.
- Caso o valor retornado seja falso, o corpo do *if* é pulado.

# if

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

void main( ){
    int num_magico, x;
    num_magico = rand();
    printf("Digite um número: ");
    scanf("%d", &x);
    if(x == num_magico)
        printf("Parabéns, você acertou o número
mágico!!\n");
}
```

# if-else

```
if (condição)
    comando;
else
    outrocomando;
```

```
if (condição){
    bloco de comandos;
}
else {
    outrocomando;
}
```

- Se a condição for verdadeira, o comando ou bloco de comandos que segue o *if* é executado.
- Caso o valor retornado seja falso, o comando que segue o *else* é executado.
- Cada comando *else* refere-se ao último *if* aberto.

# if-else

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

void main( ){
    int num_magico, x;
    num_magico = rand();
    printf("Digite um número: ");
    scanf("%d", &x);
    if(x == num_magico)
        printf("Parabéns, você acertou o número
mágico!!\n");
    else
        printf("Infelizmente você errou!!!\n");
}
```

# A escada if-else if

```
if (condição)
    comando;
else if(condição)
    outrocomando;
else if(condição)
    maisumcomando;
...
else
    comando;
```

# A escada if-else if

- Assim que uma condição verdadeira é encontrada, o comando associado a ela é executado e os demais comandos são desviados.
- Se todas as condições forem falsas, o último *else* é executado.
- Caso não exista o último *else* e todas as condições forem falsas, nenhuma ação ocorrerá.



# A escada if-else if

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

void main( ){
    int num_magico, x;
    num_magico = rand();
    printf("Digite um número: ");
    scanf("%d", &x);
    if(x == num_magico)
        printf("Parabéns, você acertou o número mágico!!\n");
    else if(x > num_magico)
        printf("O número digitado é muito alto!!!\n");
    else
        printf("O número digitado é muito baixo!!!\n");
}
```

# O operador ternário ?:

$\text{Exp1} ? \text{Exp2} : \text{Exp3}$

- Se Exp1 é verdadeira, o resultado da expressão é Exp2.
- Se Exp1 é falsa, o resultado da expressão é Exp3.

# O operador ternário ?:

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

void main( ){
    int a, b, maior, menor;
    printf("Digite dois números inteiros: ");
    scanf("%d %d", &a, &b);
    maior = (a>b) ? a : b;
    menor = (a<b) ? a : b;
    printf("%d é maior e %d é menor.\n", maior, menor);
}
```

# O operador ternário ?:

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

void main( ){
    int a, b;
    printf("Digite dois números inteiros: ");
    scanf("%d %d", &a, &b);
    printf("%d é maior.\n", (a>b) ? a : b);
}
```

# A expressão condicional

- Utiliza-se qualquer comando válido em C para controlar o *if* ou o operador `?:`.
- Não é necessário utilizar apenas comandos envolvendo os operadores relacionais e lógicos.

# A expressão condicional

```
#include <stdio.h>

void main( ){
    int a, b;

    printf("Digite dois números: ");
    scanf("%d %d", &a, &b);

    if(b)
        printf("Resultado da divisão: %d\n", a/b);
    else
        printf("Não existe divisão por zero!\n");
}
```

# switch

```
switch(expressão){  
    case constante1:  
        comandos;  
        break;  
    case constante2:  
        comandos;  
        break;  
    ...  
    default:  
        comandos;  
}
```

# switch

- Valor da expressão testado, na ordem, contra os valores das constantes especificados nos comandos *case*.
- Quando encontrar coincidência, os comandos associados ao *case* são executados.
- Comando *default* é executado se nenhuma coincidência for detectada.
- *Default* é opcional.
- Comando *switch*, diferente do *if*, só pode testar igualdade.
- Duas constantes *case* no mesmo *switch* não podem ter valores idênticos.
- Constantes de caracteres em um *switch* são automaticamente convertidas para seus valores inteiros.



```
#include <stdio.h>
void main( ){
    float oper1, oper2;
    char op;

    printf("Digite: operando1 operador operando2\n");
    scanf("%f %c %f", &oper1, &op, &oper2);
    switch(op) {
        case '+': printf("%f+%f = %f\n", oper1, oper2, oper1+oper2);
                    break;
        case '-': printf("%f-%f = %f\n", oper1, oper2, oper1-oper2);
                    break;
        case '*': printf("%f*%f = %f\n", oper1, oper2, oper1*oper2);
                    break;
        case '/': printf("%f/%f = %f\n", oper1, oper2, oper1/oper2);
                    break;
        default: printf("Operador %c desconhecido!\n", op);
    }
}
```

# switch

- Se o comando *break* é omitido, a execução continua pelos próximos comandos *case* até que um *break* ou o fim do *switch* seja encontrado.

```
switch(expressão){  
    case constante1:  
        comandos;  
    case constante2:  
        comandos;  
    case constante3:  
        comandos;  
        break;  
}
```

Escreva um programa, utilizando a expressão condicional **switch**, que calcule a área de um círculo, quadrado ou triângulo. O programa deverá ler do usuário uma letra que deverá ser ou C (para calcular a área de um círculo), ou Q (para calcular a área de um quadrado), ou T (para calcular a área de um triângulo). Conforme a figura geométrica desejada, o programa deverá solicitar do usuário a informação necessária para efetuar o cálculo da área correspondente. Por exemplo, para um círculo o usuário deve informar o raio do círculo; para um quadrado o usuário deve informar o lado do quadrado e para um triângulo o usuário deve informar o valor da base e altura do triângulo.

# Exercícios

- Ler dois números quaisquer e dividir o primeiro pelo segundo evitando divisão por zero.
- Ler dois números inteiros e exibir o maior deles.
- Ler um número inteiro e imprimir se ele é par ou ímpar.
- Ler três números inteiros e exibir o maior deles.

# Exercícios

- Fazer um algoritmo que leia 3 notas de um aluno (tipo float) e imprima a mensagem de aluno aprovado ou reprovado conforme o seguinte critério: se a média final do aluno (média aritmética simples) for maior ou igual a 6 (seis) e nenhuma das notas for inferior a 5 (cinco), o aluno foi aprovado, caso contrário o aluno foi reprovado. Use operadores lógicos (&&, ||, etc...) para diminuir o número de comandos IF...ELSE utilizados.