

Estruturas de Seleção

C3

Universidade Federal de Rio Grande
Centro de Ciências Computacionais

Operadores Relacionais

Operador	Ação
$>$	Maior do que
\geq	Maior ou igual a
$<$	Menor do que
\leq	Menor ou igual a
$==$	Igual a
\neq	Diferente de

Operadores Relacionais

- ▶ Os operadores relacionais retornam verdadeiro (1) ou falso (0).
- ▶ Exemplo:

```
#include <stdio.h>
int main() {
    int i, j;
    printf("\nEntre com dois numeros inteiros: ");
    scanf("%d%d", &i, &j);
    printf("\n%d == %d eh %d\n", i, j, i==j);
    printf("\n%d != %d eh %d\n", i, j, i!=j);
    printf("\n%d <= %d eh %d\n", i, j, i<=j);
    printf("\n%d >= %d eh %d\n", i, j, i>=j);
    printf("\n%d < %d eh %d\n", i, j, i<j);
    printf("\n%d > %d eh %d\n", i, j, i>j);
    return(0);
}
```

Operadores Lógicos

Operador	Ação
&&	AND (E)
	OR (OU)
!	NOT (NÃO)

Operadores Lógicos

► Exemplo:

```
#include <stdio.h>
int main() {
    int i, j;
    printf("informe dois numeros(0 ou 1): ");
    scanf("%d%d", &i, &j);
    printf("%d AND %d eh %d\n", i, j, i && j);
    printf("%d OR %d eh %d\n", i, j, i || j);
    printf("NOT %d eh %d\n", i, !i);
}
```

Operadores Lógicos Bit a Bit

- ▶ A Ling. C permite fazer operações bit a bit.
- ▶ Essas operações são normalmente utilizadas em aplicações de baixo nível (interação direta com o *hardware*).

Operador	Ação
&	AND
	OR
^	XOR (OR exclusivo)
~	NOT
>>	Deslocamento de bits a direita
<<	Deslocamento de bits a esquerda

Estruturas de Controle

- ▶ Estruturas de controle alteram o fluxo de execução de um programa.
 - ▶ Estruturas de Seleção: `if...else` e `switch`
 - ▶ Estruturas de Repetição: `while`, `do...while` e `for`

Estrutura de Seleção - if

- ▶ Executar uma instrução somente se um condição for satisfeita.
- ▶ Sintaxe:

```
if (condicao)
    instrucao;
```

- ▶ Note que “dentro” do if, apenas uma instrução é permitida.
- ▶ Mas e se quisermos colocar mais instruções? Usamos blocos de instruções: { }

```
if (condicao) {
    instrucao_1;
    instrucao_2;
    ...
    instrucao_n;
}
```


Exemplo 01- if

```
#include <stdio.h>
int main(){
    int A, B, C;
    printf("Digite tres numeros");
    scanf("%d %d %d",&A, &B, &C);
    if (A + B > C)
        printf("A+B eh maior que C. \n");
    printf("Esse print estah FORA do if.\n");
    return 0;
}
```

Exemplo 02 - if

```
#include <stdio.h>
int main(){
    int A, B, C;
    printf("Digite tres numeros");
    scanf("%d %d %d",&A, &B, &C);
    if (A + B > C) {
        printf("A+B eh maior que C. \n");
        printf("Esse print estah DENTRO do if.\n");
    }
    return 0;
}
```

Estrutura de Seleção - if...else

- ▶ Executar uma instrução ou outra, dependendo da condição.
- ▶ Sintaxe:

```
if (condicao)
    instrucao;
else
    instrucao;
```

Exemplo 01- if...else

```
#include <stdio.h>
int main(){
    int n;
    printf("Digite um numero inteiro: ");
    scanf("%d",&n);
    if ((n % 2) == 0)
        printf("O numero digitado eh par.\n");
    else
        printf("O numero digitado eh impar.\n");
    return 0;
}
```

Estrutura de Seleção - if...else

- ▶ Executar um conjunto de instruções ou outro, dependendo da condição.
- ▶ Sintaxe:

```
if (condicao) {  
    instrucao_1;  
    instrucao_2;  
    ...  
    instrucao_n;  
}  
else {  
    instrucao_1;  
    instrucao_2;  
    ...  
    instrucao_n;  
}
```

Exemplo 02- if...else

```
#include <stdio.h>
int main(){
    int a,b,r;
    printf("Digite dois numeros: ");
    scanf("%d %d",&a,&b);
    if (b==0){
        printf("Erro!!");
        printf("Impossivel fazer divisao por zero.\n");
    }
    else {
        r=a/b;
        printf("a/b = %d \n",r);
    }
    return 0;
}
```

ifs Aninhados

- ▶ Em C é possível colocar uma estrutura de if dentro de outra.
- ▶ Exemplo:

```
if (condicao1) {  
    Instrucoes  
}  
else{  
    if (condicao2) {  
        Instrucoes  
    }  
    else{  
        Instrucoes  
    }  
}
```

Exemplo - Ordena Três Números

```
#include<stdio.h>
int main() {
    int a,b,c;
    printf("Entre com tres numeros: ");
    scanf("%d %d %d",&a,&b,&c);
    if (a>b && a>c)
        if (b>c)
            printf("\n %d > %d > %d \n",a,b,c);
        else
            printf("\n %d > %d > %d \n",a,c,b);
    else
        if(b>a && b>c)
            if (a>c)
                printf("\n %d > %d > %d \n",b,a,c);
            else
                printf("\n %d > %d > %d \n",b,c,a);
        else
            if(a>b)
                printf("\n %d > %d > %d \n",c,a,b);
            else
                printf("\n %d > %d > %d \n",c,b,a);
    return 0;
}
```


Exercício

- ▶ Faça um programa que calcule a média de três notas de um aluno e classifique o aluno da seguinte forma:
 - ▶ Aprovado, se média maior ou igual a 5.0
 - ▶ Exame, se média maior ou igual a 3.0 e menor que 5.0
 - ▶ Reprovado, se média menor que 3.0

Exercício - Solução

```
#include<stdio.h>
int main() {
    int nota1,nota2,nota3;
    float media;
    printf("Entre com tres notas: ");
    scanf("%d %d %d",&nota1,&nota2,&nota3);
    media = (float)(nota1+nota2+nota3)/3.0;
    if (media>=5)
        printf("Aprovado !! \n");
    else
        if (media>=3)
            printf(" Exame !! \n");
        else
            printf(" Reprovado !! \n");
    return 0;
}
```

Múltipla Escolha

- ▶ Algumas vezes, em um problema é necessário escolher uma entre várias alternativas.
- ▶ Para esses casos, a linguagem C oferece a estrutura de switch.
- ▶ Sintaxe:

```
switch (var) { // Var tem que ser int ou char
    case Valor_1: // Instrucoes
        break;
    case Valor_2: // Instrucoes
        break;
    ...
    case Valor_n: // Instrucoes
        break;
    //Opcional
    default: //Instrucoes
}
```

Exemplo - Menu

```
#include <stdio.h>
int main(){
    int opcao;
    printf("\n ----Menu---- ");
    printf("\n      1- File de Frango");
    printf("\n      2- Bife a parmegiana");
    printf("\n      3- Frango a passarinho");
    printf("\n      4- Viradinho paulista");
    printf("\n      5- Salada selvagem");
    printf("\n      6- Suco");
    printf("\n ----- \n   opcao: ");
    scanf("%d",&opcao);
    printf("\n\n Voce escolheu ");
    switch (opcao) {
        case 1:
            printf("File de Frango");
            break;
        case 2:
            printf("Bife a parmegiana");
            break;
```

```
        case 3:
            printf("Frango a passarinho");
            break;
        case 4:
            printf("Viradinho paulista");
            break;
        case 5:
            printf("Salada selvagem");
            break;
        case 6:
            printf("Suco");
            break;
        default:
            printf("uma opcao invalida.");
    }

    return 1;
}
```

Comando break

- ▶ Interrompe o comando `switch` (ou qualquer outro comando).
- ▶ É comum que cada `case` tenha um `break`, mas não é obrigatório.

Exemplo - Vogal ou Consoante

```
#include <stdio.h>
int main(void){
    char letra;
    printf("digite uma letra (minuscula)");
    scanf("%c",&letra);
    switch(letra){
        case 'a': printf("Vogal"); break;
        case 'e': printf("Vogal"); break;
        case 'i': printf("Vogal"); break;
        case 'o': printf("Vogal"); break;
        case 'u': printf("Vogal"); break;
        default: printf("Consoante"); break;
    }
    return 1;
}
```

Exemplo - Vogal ou Consoante - Versão 2

```
#include <stdio.h>
int main(void){
    char letra;
    printf("digite uma letra (minuscula)");
    scanf("%c",&letra);
    switch(letra){
        case 'a':
        case 'e':
        case 'i':
        case 'o':
        case 'u':
            printf("Vogal");
            break;
        default:
            printf("Consoante");
            break;
    }
    return 1;
}
```

Exemplo - Calculadora

```
#include <stdio.h>
int main(){
    int erro=0;
    float operando1,operando2,resultado;
    char operador;

    printf("\n conta: ");
    scanf("%f %c %f",
          &operando1,
          &operador,
          &operando2);

    switch (operador) {
    case '+':
        resultado = operando1 + operando2;
        break;
    case '-':
        resultado = operando1 - operando2;
        break;
    case '*':
        resultado = operando1 * operando2;
```

```
        break;
    case '/':
        if (operando2 !=0)
            resultado = operando1 / operando2;
        else
            erro=1;
        break;
    default:
        erro = 1;
    }

    if (erro)
        printf(" Erro !! ");
    else
        printf("\n\n %.2f %c %.2f = %.2f \n\n",
              operando1,operador,
              operando2,resultado);

    return 1;
}
```


Exercício

- ▶ Escreva um programa que leia um número e mostre se esse é par e está entre 0 e 10, se ele é ímpar e está entre 0 e 10 ou se ele não está no intervalo entre 0 e 10. Utilize o comando switch.

Exercício - Solução

```
/* Solucao utilizando apenas o comando switch */
#include <stdio.h>
int main(void){
    int num;
    printf("digite um numero: ");
    scanf("%d",&num);
    switch(num){
        case 1:
        case 3:
        case 5:
        case 7:
        case 9:
            printf("Impar e esta entre 0 e 10");
            break;
        case 2:
        case 4:
        case 6:
        case 8:
        case 10:
            printf("Par e esta entre 0 e 10");
            break;
        default:
            printf("Nao esta entre 0 e 10.");
            break;
    }
    return 1;
}
```

Exercício

- Uma empresa vai dar um aumento aos seus profissionais de forma proporcional a sua categoria, portanto faça um programa que leia a categoria do funcionário e o salário atual e calcule qual será o novo salário do funcionário.

<i>Categoria</i>	<i>Aumento</i>
A	40%
B, C e D	30%
E e F	20%
G e H	10%
restante	5%

Exercício - Solução

```
/* Solucao utilizando o comando switch */
#include <stdio.h>
int main(void){
    char cat;
    float sal,novo_sal;

    printf("\n Categoria...: ");
    scanf(" %c",&cat);
    printf("\n Salario.....: ");
    scanf("%f",&sal);

    switch(cat){
    case 'A':
        novo_sal=sal*1.4;
        break;

    case 'B':
    case 'C':
    case 'D':
        novo_sal=sal*1.3;
        break;
```

```
    case 'E':
    case 'F':
        novo_sal=sal*1.2;
        break;

    case 'G':
    case 'H':
        novo_sal=sal*1.1;
        break;

    default:
        novo_sal=sal*1.05;
        break;
    }
    printf("\n O novo salario eh %f \n\n",
        novo_sal);
    return 1;
}
```

Referências

- ▶ ASCENCIO, Ana F. G.; CAMPOS, Edilene A. V. Fundamentos da programação de computadores. Pearson Prentice Hall, 2007.
- ▶ Schildt, Herbert. C completo e total. Pearson Makron Books, 1997.
- ▶ Notas de aula do Prof. Flavio Keidi Miyazawa.
- ▶ Notas de aula do Prof. Emanuel Estrada.
- ▶ Notas de aula do Prof. Alessandro Bicho.