

Comandos de Controle de Programa

Prof. Debora Medeiros

Baseado no material de:
Ciro Trindade (Unisantos), Roseli Romero (USP) e Renata Fortes (USP)

Comandos de Seleção: *if*

if (expressão é verdadeira)

execute comando ou bloco de comandos ;

else /* se expressão é falsa */

execute comando ou bloco de comandos ;

Ex:

```
if ( count > 9 )
    count = 0;
else
    count++;
```

2

Comandos de Seleção: aninhamento de *if*

- É possível aninhar construções do tipo *if-else* em diversos níveis

```
if (cond1) /* if1 */
  if (cond2) /* if2 */
    comando if2;
  else /* else2 */
    comando else2;
else /* else1 */
  if (cond3) /* if3 */
    if (cond4) /* if4 */
      comando if4;
    else /* else4 */
      comando else4;
  else /* else3 */
    comando else3;
```

O *if* aninhado é simplesmente um *if* dentro da declaração de um outro *if* externo. O único cuidado que devemos ter é o de saber exatamente a qual *if* um determinado *else* está ligado.

3

Comandos de Seleção: aninhamento de *if*

- Correspondência entre *if*'s e *else*'s

```
if (cond1)
  if (cond2)
    comando if2;
  else /* atenção: else2! */
    comando if1; /* erro: comando do if2 */

modo correto:

if (cond1) {
  if (cond2)
    comando if2;
}
else
  comando if1;
```

4

Comandos de Seleção: aninhamento de *if*

Salário	Desconto do INSS
Até R\$ 1.024,97	8,00%
De R\$ 1.024,98 a R\$ 1.708,27	9,00%
Acima de R\$ 1.708,27	11,00%

```
#include <stdio.h>
int main() {
    float salario, inss;

    printf("Informe o salario do funcionario: R$ ");
    scanf("%f", &salario);

    if (salario < 1024.98) {
        inss = salario * 0.08;
    }
    else {
        if (salario <= 1708.27) {
            inss = salario * 0.09;
        }
        else {
            inss = salario * 0.11;
        }
    }

    printf("\nINSS: R$ %.2f\n", inss);
    system("pause");
    return 0;
}
```

Modificador de formatação
Forma geral: `%.nf` onde `n` é o número de casas decimais desejado.

5

Comandos de Seleção: aninhamento de *if*

Salário	Desconto do INSS
Até R\$ 1.024,97	8,00%
De R\$ 1.024,98 a R\$ 1.708,27	9,00%
Acima de R\$ 1.708,27	11,00%

```
#include <stdio.h>
int main() {
    float salario, inss;

    printf("Informe o salario do funcionario: R$ ");
    scanf("%f", &salario);

    if (salario < 1024.98) {
        inss = salario * 0.08;
    }
    else {
        if (salario <= 1708.27) {
            inss = salario * 0.09;
        }
        else {
            inss = salario * 0.11;
        }
    }

    printf("\nINSS: R$ %.2f\n", inss);
    system("pause");
    return 0;
}
```

```
Informe o salario do funcionario: R$ 1000
INSS: R$ 80.00
Press any key to continue . . .
```

```
Informe o salario do funcionario: R$ 2000
INSS: R$ 220.00
Press any key to continue . . .
```

Modificador de formatação
Forma geral: `%.nf` onde `n` é o número de casas decimais desejado.

5

```

1 int main() {
2     float n1, n2; // operandos
3     char op; // operador
4
5     printf("Calculadora Basica\n\nDigite numero operador numero:");
6     scanf("%f %c %f",&n1,&op,&n2);
7
8     if (op == '+') {
9         printf("Resultado = %.2f\n",n1 + n2);
10    }
11    else {
12        if (op == '-') {
13            printf("Resultado = %.2f\n",n1 - n2);
14        }
15        else {
16            if ((op == 'x') || (op == '*')) {
17                printf("Resultado = %.2f\n",n1 * n2);
18            }
19            else {
20                if (op == '/') {
21                    if (n2 != 0) {
22                        printf("Resultado = %.2f\n",n1 / n2);
23                    }
24                    else {
25                        printf("Erro: divisao por zero\n");
26                    }
27                }
28                else {
29                    printf("Operador invalido\n");
30                }
31            }
32        }
33    }
34    system("pause");
35    return 0;
36 }

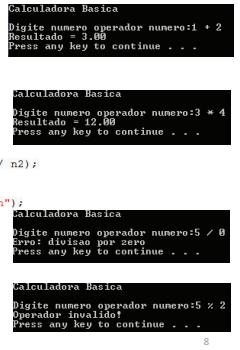
```

7

```

1 int main() {
2     float n1, n2; // operandos
3     char op; // operador
4
5     printf("Calculadora Basica\n\nDigite numero operador numero:");
6     scanf("%f %c %f",&n1,&op,&n2);
7
8     if (op == '+') {
9         printf("Resultado = %.2f\n",n1 + n2);
10    }
11    else {
12        if (op == '-') {
13            printf("Resultado = %.2f\n",n1 - n2);
14        }
15        else {
16            if ((op == 'x') || (op == '*')) {
17                printf("Resultado = %.2f\n",n1 * n2);
18            }
19            else {
20                if (op == '/') {
21                    if (n2 != 0) {
22                        printf("Resultado = %.2f\n",n1 / n2);
23                    }
24                    else {
25                        printf("Erro: divisao por zero\n");
26                    }
27                }
28                else {
29                    printf("Operador invalido\n");
30                }
31            }
32        }
33    }
34    system("pause");
35    return 0;
36 }

```



8

Comandos de Seleção: *else*

- A expressão da condição :
 - Se ela for diferente de zero
 - a seqüência_comandos_1 será executada.
 - Se for zero
 - seqüência_comandos_2 será executada.

Quando o compilador avalia uma condição, ele quer um valor de retorno para poder tomar a decisão. Mas esta expressão não necessita ser uma expressão no sentido convencional. Uma variável sozinha pode ser uma "expressão" e esta retorna o seu próprio valor. Assim:

```

int num;
...
if (num)
...
if (!num)
...

```

```

if (condição) {
    seqüência_de_comandos_1;
}
else {
    seqüência_de_comandos_2;
}

```

11

Comandos de Seleção: encadeamento *if-else-if*

```

if (condição_1) {
    seqüência_de_comandos_1;
}
else if (condição_2) {
    seqüência_de_comandos_2;
}
...
else if (condição_n) {
    seqüência_de_comandos_n;
}
else {
    seqüência_de_comandos_default;
}

```

- No encadeamento apenas *um dos n* comandos será executado: o primeiro cujo teste for verdadeiro

10

Comandos de Seleção: operador ?:

- Operador condicional ternário (?):
 - O ? : é chamado operador condicional ternário porque requer três operandos
 - Forma geral:
 - Exp1 ? Exp2 : Exp3;**
 - A **Exp1** é avaliada primeiro
 - Se seu valor for diferente de zero (verdadeiro), a **Exp2** é avaliada e seu resultado será o valor da expressão
 - Se **Exp1** for zero (falso), a **Exp3** é avaliada e será o valor da expressão

11

Comandos de Seleção: operador ?:

```

if (a > 0)
    b = -150;
else
    b = 150;

```

Simplificação usando o operador ?:
`b = a > 0 ? -150 : 150;`

```

exp1 ? exp2 : exp3
if (exp1)
    exp2;
else
    exp3;

```

```

z = max(x, y);
if: if (x > y)
    z=x;
else
    z=y;
?: z = (x > y) ? x : y;

```

```

x % 2 == 0 ? printf("Par") : printf("Ímpar");

```

12

Comandos de Seleção: switch

- Usado no lugar de um bloco de if-else-if...
- Testa múltiplas opções

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

Testa a igualdade entre as constantes e a expressão

Termina a execução do switch

13

Exemplo da calculadora usando switch

```
1 /* Este simula uma calculadora com as 4 operações básicas */
2 #include <stdio.h>
3 int main() {
4     float n1, n2; // operandos
5     char op; // operador
6     printf("Calculadora Basica\n\nDigite numero operador numero:");
7     scanf("%f %c %f", &n1, &op, &n2);
8     switch(op) {
9         case '+':
10            printf("Resultado = %.2f\n", n1 + n2);
11            break;
12         case '-':
13            printf("Resultado = %.2f\n", n1 - n2);
14            break;
15         case '*':
16            printf("Resultado = %.2f\n", n1 * n2);
17            break;
18         case '/':
19            if(n2 != 0) {
20                printf("Resultado = %.2f\n", n1 / n2);
21            }
22            else {
23                printf("Erro: divisao por zero\n");
24            }
25            break;
26         default:
27            printf("Operador invalido\n");
28     } // fim do switch
29     system("pause");
30     return 0;
31 } // fim da função main()
```

14

Permitindo x ou *

```
1 /* Este simula uma calculadora com as 4 operações básicas */
2 #include <stdio.h>
3 int main() {
4     float n1, n2; // operandos
5     char op; // operador
6     printf("Calculadora Basica\n\nDigite numero operador numero:");
7     scanf("%f %c %f", &n1, &op, &n2);
8     switch(op) {
9         case '+':
10            printf("Resultado = %.2f\n", n1 + n2);
11            break;
12         case '-':
13            printf("Resultado = %.2f\n", n1 - n2);
14            break;
15         case 'x':
16         case '*':
17            printf("Resultado = %.2f\n", n1 * n2);
18            break;
19         case '/':
20            if(n2 != 0) {
21                printf("Resultado = %.2f\n", n1 / n2);
22            }
23            else {
24                printf("Erro: divisao por zero\n");
25            }
26            break;
27         default:
28            printf("Operador invalido\n");
29     } // fim do switch
30     system("pause");
31     return 0;
32 } // fim da função main()
```

15

Comandos de Repetição

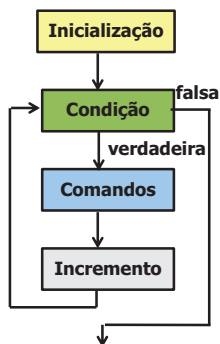
- O Laço for
- Muito flexível e poderoso
- Forma geral

```
for(inicialização; condição; incremento) {
    sequência de comandos;
}
```

- inicialização:** geralmente uma atribuição usada para definir a variável de controle do laço
- condição:** expressão que determina quando o laço terminará
- incremento:** define como a variável de controle do laço será modificada cada vez que o laço é repetido

16

Fluxo de execução do for



17

Exemplo for

- Imprimir os n primeiros termos da sequência $Y_{k+1} = Y_k + 2$

```
1 #include <stdio.h>
2 int main() {
3     int x, n, y = 1;
4
5     printf("Digite o número de termos (n): ");
6     scanf("%d", &n);
7     for(x = 1; x <= n; x++) {
8         printf("%4d", y);
9         y += 2;
10    }
11    printf("\n");
12    system("pause");
13    return 0;
14 }
```

Modificador de formatação
Forma geral: %nd onde n especifica um campo mínimo para a saída e a posiciona com espaços para preservar o tamanho desejado.

```

Digite o número de termos (n): 4
1 3 5 7
Press any key to continue . . .
  
```

18