



Aula: Condicionais (if – else / switch)

Introdução a Programação

Túlio Toffolo & Puca Huachi
<http://www.toffolo.com.br>

Departamento de Computação
Universidade Federal de Ouro Preto

Aula: Condicionais (if – else / switch)

- 1 Exemplos (if)
- 2 Condicionais (if – else)
- 3 Exercícios
- 4 Comandos de decisão aninhados
- 5 Comandos de decisão: switch
- 6 Exemplos
- 7 Exercícios

Aula de Hoje

- 1 Exemplos (if)
- 2 Condicionais (if - else)
- 3 Exercícios
- 4 Comandos de decisão aninhados
- 5 Comandos de decisão: switch
- 6 Exemplos
- 7 Exercícios

Exemplo usando o comando `if`

Exemplo 1 (resolvido anteriormente):

Faça um programa em C/C++, para calcular a área de um círculo. A área de um círculo é dada pela seguinte fórmula $a = \pi r^2$. O valor do raio r será digitado pelo usuário.

- Altere o programa anterior para calcular a área somente se o valor do raio for positivo.

```

1  /* Programa que calcula a área de um círculo
2  */
3
4  #include <stdio.h>
5
6  int main()
7  {
8      // declaração da constante Pi
9      const double PI = 3.141592;
10     double raio;
11
12     printf("Digite o raio do círculo: ");
13     scanf("%lf", &raio);
14
15     // calculando e imprimindo a área do círculo
16     if (raio >= 0) {
17         double area = PI * raio * raio;
18         printf("\nÁrea do círculo: %lf\n", area);
19     }
20
21     return 0;
22 }

```

Exemplo 2

Faça um programa em C/C++, para calcular a área de um círculo. A área de um círculo é dada pela seguinte fórmula $a = \pi r^2$. O valor do raio r será digitado pelo usuário.

Verifique se o raio é positivo antes de efetuar cálculo, **caso contrário imprima uma mensagem de erro ao usuário.**

```
1  /* Programa que calcula a área de um círculo
2   */
3
4  #include <stdio.h>
5
6  int main()
7  {
8      // declaração da constante Pi
9      const double PI = 3.141592;
10     double raio;
11
12     printf("Digite o raio do círculo: ");
13     scanf("%lf", &raio);
14
15     // calculando e imprimindo a área do círculo
16     if (raio >= 0) {
17         double area = PI * raio * raio;
18         printf("\nÁrea do círculo: %lf\n", area);
19     }
20     // caso o raio seja negativo, imprime mensagem de erro
21     if (raio < 0)
22         printf("Erro: valor do raio é negativo...\n");
23
24     return 0;
25 }
```

Exemplo 3

Codifique um programa que lê um número inteiro positivo. A seguir o programa deve imprimir uma mensagem para o usuário dizendo se o número digitado é **par ou impar**.


```

1 // Programa que verifica se um no. é par ou impar
2 #include <stdio.h>
3
4 int main()
5 {
6     int numero; //variável para armazenar o número
7     printf("Digite um numero inteiro: ");
8     scanf("%d", &numero);
9
10    // testa se o número é par
11    if (numero % 2 == 0) {
12        printf("\n0 número %d é par.\n", numero);
13    }
14
15    // testa se o número é impar
16    if (numero % 2 != 0) {
17        printf("\n0 número %d é impar\n.", numero);
18    }
19
20    return 0;
21 }

```

Exemplo de execução

- Execução 1:

Digite um numero inteiro: 5

O número 5 é ímpar

- Execução 2:

Digite um numero inteiro: 8

O número 8 é par

Aula de Hoje

- 1 Exemplos (if)
- 2 Condicionais (if - else)
- 3 Exercícios
- 4 Comandos de decisão aninhados
- 5 Comandos de decisão: switch
- 6 Exemplos
- 7 Exercícios

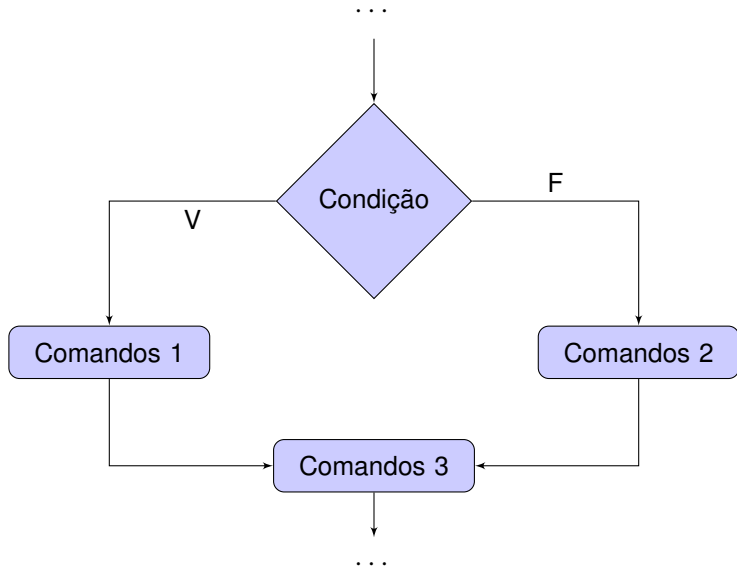
O Comando `if`

- Nos exercícios anteriores, o programa teve que tomar decisões baseadas nos dados fornecidos.
- Para tal, utilizamos dois comandos `if`.
- No entanto, podemos (e devemos) simplificar os programas com o uso do comando `if-else`.

O Comando `if-else`

- O comando `if` permite executar um ou mais comandos se a expressão relacional resultar em **verdadeiro**. Se desejarmos que algo seja executado se a expressão relacional resultar em **falso**, então devemos utilizar o comando `if-else`.
- Dessa forma, podemos executar um ou mais comandos se o teste for **verdadeiro**; ou executar um ou mais comandos distintos se o teste for **falso**.
- O comando `if-else` consiste no comando `if` seguido de uma instrução (ou um bloco de instruções), seguido da palavra-chave `else`, que por sua vez é seguida de uma instrução (ou um bloco de instruções).

Tomada de decisão



Sintaxe do comando if-else

```
1  if ( <expressão_de_teste> )
2      instrução_única_V;
3  else
4      instrução_única_F;
```

OU

```
1  if ( <expressão_de_teste> )
2  {
3      instrução_V1;
4      ...
5      instrução_Vn;
6  }
7  else
8  {
9      instrução_F1;
10     ...
11     instrução_Fn;
12 }
```

Exemplo 1

- Faça um programa em C/C++, para calcular a área de um círculo. A área de um círculo é dada pela seguinte fórmula $a = \pi r^2$. O valor do raio r será digitado pelo usuário.

Verifique se o raio é positivo antes de efetuar cálculo, caso contrário imprima uma mensagem de erro ao usuário. **Utilize if-else.**


```
1  #include <stdio.h>
2
3  int main()
4  {
5      // declaração da constante Pi
6      const double PI = 3.141592;
7      double raio;
8
9      printf("Digite o raio do círculo: ");
10     scanf("%lf", &raio);
11
12     // se raio for positivo, calcula e imprime a área do círculo
13     if (raio >= 0) {
14         double area = PI * raio * raio;
15         printf("\nÁrea do círculo: %lf\n", area);
16     }
17     // caso contrário, imprime mensagem de erro
18     else {
19         printf("Erro: valor do raio é negativo...\n");
20     }
21
22     return 0;
23 }
```

Exemplo 2

- Codifique um programa que leia um número inteiro positivo. A seguir o programa imprime uma mensagem para o usuário dizendo se o número digitado é par ou impar. Utilize `if-else`.

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int numero; //variável para armazenar o número
6
7      printf("Digite um numero inteiro: ");
8      scanf("%d", &numero);
9
10     // se o número for par...
11     if (numero % 2 == 0)
12         printf("\nO número %d é par.\n", numero);
13
14     // caso contrário
15     else
16         printf("\nO número %d é ímpar.\n", numero);
17
18     return 0;
19 }
```

Exemplo 3

- Escreva um programa para encontrar as raízes de uma equação do segundo grau:

$$ax^2 + bx + c = 0$$

- Os coeficientes da equação são reais. O programa faz a alocação de 3 posições de memória para esses coeficientes, inicializando-os com o valor zero.
- O programa efetua a leitura dos coeficientes através do teclado e, em seguida, calcula o valor das raízes existentes.
- Caso não existam raízes reais, o programa deve informar este fato ao usuário.

```
1  #include <stdio.h>
2  #include <math.h>
3
4  int main()
5  {
6      double a, b, c, delta;
7
8      printf("\nCoeficiente a: "); scanf("%lf", &a);
9      printf("\nCoeficiente b: "); scanf("%lf", &b);
10     printf("\nCoeficiente c: "); scanf("%lf", &c);
11     delta = pow(b, 2) - 4 * a * c;
12
13     if (delta < 0) {
14         printf("\nNão existem raízes reais\n");
15     }
16     else {
17         double x1 = (-b + sqrt(delta)) / (2 * a);
18         double x2 = (-b - sqrt(delta)) / (2 * a);
19         printf("\nX1 = %lf\n", x1);
20         printf("X2 = %lf\n", x2);
21     }
22     return 0;
23 }
```

Exemplo 4

- Como melhorar o programa anterior para informar o usuário as situações a seguir?
 - Não existem raízes reais ($\Delta < 0$);
 - Existem raízes reais idênticas ($\Delta = 0$);
 - Existem raízes reais distintas ($\Delta > 0$);

```

1  int main()
2  {
3      double a, b, c, delta;
4
5      printf("\nCoeficiente a: "); scanf("%lf", &a);
6      printf("\nCoeficiente b: "); scanf("%lf", &b);
7      printf("\nCoeficiente c: "); scanf("%lf", &c);
8      delta = pow(b, 2) - 4 * a * c;
9
10     if (delta < 0) {
11         printf("\nNão existem raízes reais\n");
12     }
13     else {
14         if (delta == 0) {
15             double x = (-b + sqrt(delta)) / (2 * a);
16             printf("\nRaízes idênticas: X1 = X2 = %lf\n", x);
17         }
18         else {
19             double x1 = (-b + sqrt(delta)) / (2 * a);
20             double x2 = (-b - sqrt(delta)) / (2 * a);
21             printf("\nRaízes distintas: X1 = %lf, X2 = %lf\n", x1, x2);
22         }
23     }
24     return 0;
25 }

```

```

1  int main()
2  {
3      double a, b, c, delta;
4
5      printf("\nCoeficiente a: "); scanf("%lf", &a);
6      printf("\nCoeficiente b: "); scanf("%lf", &b);
7      printf("\nCoeficiente c: "); scanf("%lf", &c);
8      delta = pow(b, 2) - 4 * a * c;
9
10     if (delta < 0) {
11         printf("\nNão existem raízes reais\n");
12     }
13     else if (delta == 0) {
14         double x = (-b + sqrt(delta)) / (2 * a);
15         printf("\nRaízes idênticas: X1 = X2 = %lf\n", x);
16     }
17     else {
18         double x1 = (-b + sqrt(delta)) / (2 * a);
19         double x2 = (-b - sqrt(delta)) / (2 * a);
20         printf("\nRaízes distintas: X1 = %lf, X2 = %lf\n", x1, x2);
21     }
22     return 0;
23 }

```


Aula de Hoje

- 1 Exemplos (if)
- 2 Condicionais (if - else)
- 3 Exercícios**
- 4 Comandos de decisão aninhados
- 5 Comandos de decisão: switch
- 6 Exemplos
- 7 Exercícios

Exercícios

Exercício 1

- Codifique um programa que faça a leitura de dois número reais. A seguir o programa lê um caractere, que deve ser **+**, **-**, ***** ou **/**, e realiza a operação indicada pelo caractere sobre os valores lidos.
- O programa deve imprimir os valores e o resultado da operação realizada sobre eles, como mostra o exemplo a seguir (usando exatamente uma casa decimal):

```
1 Primeiro valor: 4.5
2 Segundo valor: 5
3 Operador: +
4
5 Resultado: 4.5 + 5.0 = 9.5
```

Aula de Hoje

- 1 Exemplos (if)
- 2 Condicionais (if - else)
- 3 Exercícios
- 4 Comandos de decisão aninhados**
- 5 Comandos de decisão: switch
- 6 Exemplos
- 7 Exercícios

Comandos de decisão aninhados

Como já sabemos, podemos (e em muitos casos devemos) aninhar *ifs*:

- Ao aninhar *ifs*, aparecem os famosos `else if`
- Qual código você prefere?

```
1  if (...) {  
2      ...  
3  }  
4  else {  
5      if (...) {  
6          ...  
7      }  
8      else {  
9          if (...) {  
10             ...  
11         }  
12         else {  
13             ...  
14         }  
15     }  
16 }
```

```
1  if (...) {  
2      ...  
3  }  
4  else if (...) {  
5      ...  
6  }  
7  else if (...) {  
8      ...  
9  }  
10 else {  
11     ...  
12 }
```

Note que os dois códigos são **equivalentes!**

Exercícios

Exemplo

Crie um programa que lê o conceito de um aluno na disciplina BCC201 (Introdução à Programação) e imprime seu significado, de acordo com a tabela abaixo. Caso seja informado um conceito inexistente, deve ser exibida uma mensagem de erro.

Conceito	Significado
A	Excelente
B	Ótimo
C	Bom
D	Regular
E	Ruim
F	Nos vemos de novo ano que vem...

```

1  #include <stdio.h>
2
3  int main()
4  {
5      char conceito;
6      printf("Digite o conceito: "); scanf("%c", &conceito);
7
8      if (conceito == 'A')
9          printf("\nExcelente!\n");
10     else if (conceito == 'B')
11         printf("\nÓtimo!\n");
12     else if (conceito == 'C')
13         printf("\nBom!\n");
14     else if (conceito == 'D')
15         printf("\nRegular!\n");
16     else if (conceito == 'E')
17         printf("\nRuim!\n");
18     else if (conceito == 'F')
19         printf("\nNos vemos de novo ano que vem...\n");
20     else
21         printf("\nERRO: conceito inexistente.\n");
22
23     return 0;
24 }

```

Como melhorar o programa?

Aula de Hoje

- 1 Exemplos (if)
- 2 Condicionais (if - else)
- 3 Exercícios
- 4 Comandos de decisão aninhados
- 5 Comandos de decisão: switch**
- 6 Exemplos
- 7 Exercícios

Alternativa ao uso do `if`

- No exemplo anterior vimos que a variável `conceito` pode assumir diversos valores e que o uso dos `if` aninhados pode ser trabalhoso.
- Neste caso, temos um comando para verificar várias possibilidades para valores de uma variável ou expressão.
- Em C/C++, `switch` é este comando.

Sintaxe do switch

```
1  switch (op) {  
2  case valor1:  
3  comandos1;  
4  break;  
5  case valor2:  
6  comandos2;  
7  ...  
8  default:  
9  comandosN;  
10 }
```

Funcionamento do `switch`

- O `switch` verifica se `op` é equivalente ao `valor1`,
- se for, executa o grupo de comandos1 até que encontre um `break`.
- Se `op` não for igual a `valor1`, verifica a relação com `valor2` e assim por diante.
- Se não for nenhum dos valores, o grupo de comandosN do caso `default` é executado.

```

1  #include <stdio.h>
2
3  int main()
4  {
5      char conceito;
6      printf("Digite o conceito: "); scanf("%c", &conceito);
7
8      switch (conceito) {
9          case 'A':
10             printf("\nExcelente!\n");
11             break;
12          case 'B':
13             printf("\nÓtimo!\n");
14             break;
15          case 'C':
16             printf("\nBom!\n");
17             break;
18          case 'D':
19             printf("\nRegular!\n");
20             break;
21          case 'E':
22             printf("\nRuim!\n");
23             break;
24          case 'F':
25             printf("\nNos vemos de novo ano que vem...\n");
26             break;
27          default:
28             printf("\nERRO: conceito inexistente.\n");
29      }
30      return 0;
31 }

```

Observações sobre o `switch`

- O `switch` só permite comparar expressões com **constantes**.
- Se precisarmos comparar com variáveis ou verificar faixas de valores, devemos usar o comando `if`.
- Se não usarmos o comando **`break`** em cada case o programa continuará até o fim do bloco.

Exemplo onde não se pode usar o `switch`

Exemplo 01

Crie um programa que lê o peso e a altura de um adulto e mostra sua condição de acordo com a tabela de IMC abaixo. O cálculo do IMC é dado por **peso/altura²**

IMC em adultos	Condição física
Abaixo de 18.5	Abaixo do peso
Entre 18.5 e 25	Peso normal
Entre 25 e 30	Acima do peso
Acima de 30	Obeso

Exemplo de uso do switch

```
1  switch (x) {  
2  case 1:  
3  // se x for 1, passa por aqui  
4  case 2:  
5  // se x for 1 ou 2, passa por aqui  
6  case 3:  
7  printf("x é 1, 2 ou 3\n");  
8  break;  
9  default:  
10 printf("x não é 1, 2 ou 3\n");  
11 }
```

Aula de Hoje

- 1 Exemplos (if)
- 2 Condicionais (if - else)
- 3 Exercícios
- 4 Comandos de decisão aninhados
- 5 Comandos de decisão: switch
- 6 Exemplos**
- 7 Exercícios

Exemplo 1

Escreva um programa que leia o código de um determinado produto e mostre a sua classificação. Utilize a tabela abaixo como referência:

Código	Descrição
1	Alimento não-perecível
2	Alimento perecível
3	Vestuário
4	Limpeza

```
1 //Imprime a classificação de um produto
2
3 int main()
4 {
5     int cod;
6     printf("Digite o código: ");
7     scanf("%d", &cod);
8
9     switch (cod) {
10    case 1:
11        printf("Alimento não-perecível");
12        break;
13    case 2:
14        printf("Alimento perecível");
15        break;
16    case 3:
17        printf("Vestuário");
18        break;
19    case 4:
20        printf("Limpeza");
21        break;
22    }
23    return 0;
24 }
```

Exemplo 2

Dada uma letra, escreva na tela se essa letra é uma vogal ou consoante (considerar apenas letras minúsculas).

```
1 //Verifica se uma letra é vogal ou consoante
2
3 int main()
4 {
5     char letra;
6     printf("Digite uma letra: ");
7     scanf("%c", &letra);
8
9     switch (letra) {
10     case 'a':
11     case 'e':
12     case 'i':
13     case 'o':
14     case 'u':
15         printf("Vogal\n");
16         break;
17     default:
18         printf("Consoante\n");
19     }
20     return 0;
21 }
```

Exemplo 3

Escreva um programa que indique o número de dias existentes em um mês.

Número de dias	Mês
28	02
30	04, 06, 09, 11
31	01, 03, 05, 07, 08, 10, 12

```
1 //Imprime o número de dias do mês
2
3 int main()
4 {
5     int mes;
6     printf("Digite o número do mês: ");
7     scanf("%d", &mes);
8
9     switch (mes)
10    {
11        case 1: case 3: case 5: case 7:
12        case 8: case 10: case 12:
13            printf("0 mês tem 31 dias\n");
14            break;
15        case 4: case 6: case 9: case 11:
16            printf("0 mês tem 30 dias\n");
17            break;
18        case 2:
19            printf("0 mês tem 28 dias\n");
20            break;
21    }
22    return 0;
23 }
```

Aula de Hoje

- 1 Exemplos (if)
- 2 Condicionais (if - else)
- 3 Exercícios
- 4 Comandos de decisão aninhados
- 5 Comandos de decisão: switch
- 6 Exemplos
- 7 Exercícios**

Exercícios

Exercício 1

- Codifique um programa que faça a leitura de dois número reais. A seguir o programa lê um caractere, que deve ser **+**, **-**, ***** ou **/**, e realiza a operação indicada pelo caractere sobre os valores lidos.
- O programa deve imprimir o resultado da operação realizada sobre eles, como mostra o exemplo a seguir (exibindo exatamente duas casas decimais):
- Seu programa deve usar o comando `switch`.

```
1 Resultado: 4.50 + 5.00 = 9.50
```




Perguntas?