



PROGRAMAÇÃO A

Estrutura Condicional

INTRODUÇÃO

- Uma **expressão condicional** é uma expressão cujo valor pode ser *falso* ou *verdadeiro*. Em C, **falso** é representado por **0** e **verdadeiro** é representado por **1**.
- Uma **condição simples** é uma expressão condicional cujo valor é dado por um **operador relacional**, isto é, um operador que efetua uma comparação.
- Uma **condição composta** é uma expressão condicional formada por duas ou mais condições simples ligadas por **operadores lógicos**.
- Por exemplo, em C, as expressões $(a + 1 > 2 * b)$ e $(2 * c == 6)$ são **condições simples**.
- Supondo $a = 1$ e $b = 2$, podemos concluir que o valor da expressão $(a + 1 > 2 * b)$ é **0**, ou seja, **falso**, pois neste caso $a + 1$ **não é maior que** $2 * b$. Por outro lado, supondo $c = 3$, podemos concluir que o valor da expressão $(2 * c == 6)$ é **1**, ou seja, **verdadeiro**, pois neste caso $2 * c$ **é igual a** 6 .

OPERADORES RELACIONAIS

Operação	Linguagem C	Exemplos
Igual	==	$x == 1$ (o conteúdo da variável x é igual a 1)
Diferente	!=	$x != 1$ (o conteúdo da variável x é diferente de 1)
Menor	<	$x < 1$ (o conteúdo da variável x é menor que 1)
Menor ou igual	<=	$x <= 1$ (o conteúdo da variável x é menor ou igual a 1)
Maior	>	$x > 1$ (o conteúdo da variável x é maior que 1)
Maior ou igual	>=	$x >= 1$ (o conteúdo da variável x é maior ou igual a 1)

Em C, **operadores relacionais** não podem ser combinados em uma mesma condição simples. Para entender o porquê, suponha que temos $x=5$, $y=5$ e $z=5$. Neste caso, deveríamos esperar que a expressão $x==y==z$ fosse verdadeira, porém devido à forma como C avalia os operadores relacionais, esta expressão é falsa. Isso acontece porque o valor da expressão $x==y==z$ é determinado em duas etapas. Primeiramente se avalia $x==y$, que resulta em **verdadeiro (1)**, e depois esse resultado é comparado com o valor de z , ou seja, $1==5$, que resulta em **falso (0)**. Assim, para obter o resultado correto é necessário usar uma **condição composta**.

OPERADORES RELACIONAIS - EXERCÍCIOS

Supondo que as variáveis **X**, **Y** e **Z** possuam valores **1**, **2** e **3** respectivamente, informe se as seguintes expressões condicionais simples abaixo são verdadeiras ou falsas.

- | | |
|---------------------------------|----------------------|
| a. $X > 1$ | falso ou verdadeiro? |
| b. $Y \geq 2$ | falso ou verdadeiro? |
| c. $Z == 4$ | falso ou verdadeiro? |
| d. $(X + Y) != Z$ | falso ou verdadeiro? |
| e. $X > (Y - Z)$ | falso ou verdadeiro? |
| f. $(X+1) / 2 == 0$ | falso ou verdadeiro? |
| g. $(Y*2) < (Z/3)$ | falso ou verdadeiro? |
| h. $((X+Y) / (Z-Y) \leq (X+2))$ | falso ou verdadeiro? |
| i. $(Y \% 2) != 0$ | falso ou verdadeiro? |
| j. $(Y \% 2) == 1$ | falso ou verdadeiro? |
| k. $(6 \% Y) != 0$ | falso ou verdadeiro? |
| l. $X != Y$ | falso ou verdadeiro? |
| m. $(X + 1) != ((X+Y+1) / 2)$ | falso ou verdadeiro? |

programa_slide_4.c

OPERADORES LÓGICOS

TABELA E	TABELA OU	TABELA NÃO
$V e V = V$	$V ou V = V$	Não $V = F$
$V e F = F$	$V ou F = V$	Não $F = V$
$F e V = F$	$F ou V = V$	
$F e F = F$	$F ou F = F$	

Tabela 2 - Tabela Verdade.

Os principais operadores lógicos da linguagem C são: **&&**, **||** e **!**, que significam **e**, **ou** e **não** e são usados para **conjunção**, **disjunção** e **negação**, respectivamente.

Exemplos:

- Se e chover e relampejar, eu fico em casa.
Quando eu fico em casa?
- Se chover ou relampejar, eu fico em casa.
Quando eu fico em casa?
- Se não chover eu fico em casa.
Quando eu fico em casa?

Para mais informações sobre a construção de sentenças lógicas na linguagem C, leia o arquivo [Observações importantes sobre os operadores lógicos e a construção de sentenças em linguagem C.pdf](#) disponível no Moodle.

OPERADORES LÓGICOS

- Quando avaliados, tanto os operadores relacionais quanto os operadores lógicos, resultam em um valor falso (0) ou verdadeiro (1).
- Por exemplo, em C, a expressão $(a + 1 > 2 * b \ || \ 2 * c == 6)$ é uma **condição composta**. Supondo $a = 1$, $b = 2$ e $c = 3$, podemos concluir que o valor desta expressão é **1**, ou seja, **verdadeiro**, pois neste caso $(a + 1 > 2 * b)$ vale **0**, $(2 * c == 6)$ vale **1** e $(0 \ || \ 1)$ resulta em **1**.
- Em uma expressão condicional, os operadores aritméticos são avaliados antes dos operadores relacionais que são avaliados antes dos lógicos.
- Por exemplo, se eu quiser construir uma expressão condicional que verifique se um determinado aluno será aprovado ou não em uma disciplina, eu poderia escrever a mesma da seguinte forma: $(media \geq 6.0) \ \&\& \ (frequencia \geq 75)$. Lembrando que se você trocar o operador lógico $\&\&$ por um $\ || \$ a expressão terá outro significado.

OPERADORES LÓGICOS - EXERCÍCIOS

Supondo que as variáveis **X**, **Y**, **Z**, **A** possuam valores **1**, **2**, **3** e **2015** respectivamente, informe se as seguintes expressões condicionais simples abaixo são **verdadeiras** ou **falsas**.

- a. $(X > 1) \ \&\& \ (Y < 2)$ **falso** ou **verdadeiro?**
- b. $(Y \geq 2) \ \|\| \ (2 > 3)$ **falso** ou **verdadeiro?**
- c. $(Z == 4) \ \&\& \ (X == 1)$ **falso** ou **verdadeiro?**
- d. $((X + Y) != Z) \ \|\| \ (X == 1)$ **falso** ou **verdadeiro?**
- e. $!(X > (Y - Z))$ **falso** ou **verdadeiro?**
- f. $!(X != Y)$ **falso** ou **verdadeiro?**
- g. $!((X + 1) > 2) \ \|\| \ !((X+Y+1) / 2 == 1)$ **falso** ou **verdadeiro?**

Ano Bissexto

$((A \% 400 == 0) \ \|\| \ ((A \% 4 == 0) \ \&\& \ (A \% 100 != 0)))$ **falso** ou **verdadeiro?**

programa_slide_7.c

ESTRUTURA CONDICIONAL EM ALGORITMOS

A estrutura condicional em algoritmos pode ser **simples** ou **composta**.

Estrutura condicional simples

O **comando** só será executado se a **condição** for verdadeira. Uma condição é uma comparação que possui dois valores possíveis: **verdadeiro** ou **falso**.

SE condição
ENTÃO comando

Os **comandos 1, 2 e 3** só serão executados se a **condição** for verdadeira. As palavras **INÍCIO** e **FIM** serão necessárias apenas quando dois ou mais comandos forem executados.

SE condição **ENTÃO**
INÍCIO
 comando 1
 comando 2
 comando 3
FIM

ESTRUTURA CONDICIONAL EM ALGORITMOS

Estrutura condicional composta

Se a **condição** for verdadeira, será executado o **comando 1**; caso contrário, será executado o **comando 2**.

```
SE condição
  ENTÃO comando 1
  SENÃO comando 2
```

Se a **condição** for verdadeira, o **comando 1** e o **comando 2** serão executados; caso contrário, o **comando 3** e o **comando 4** serão executados.

```
SE condição ENTÃO
  INÍCIO
    comando 1
    comando 2
  FIM
  SENÃO
  INÍCIO
    comando 3
    comando 4
  FIM
```

ESTRUTURA CONDICIONAL EM C

Estrutura condicional simples

O **comando** só será executado se a **condição** for verdadeira. Uma condição é uma comparação que possui dois valores possíveis: **verdadeiro** ou **falso**.

```
if (condição)
    comando;
```

```
if (condição)
{
    comando1;
    comando2;
    comando3;
}
```

Importante!

Em C, torna-se obrigatória a utilização de chaves quando existe mais de um comando a executar. Os comandos entre chaves { } só serão executados se a condição for verdadeira.

ESTRUTURA CONDICIONAL EM C

Estrutura condicional composta

Se a **condição** for verdadeira, será executado o **comando1**; se for falsa, será executado o **comando2**.

```
if (condição)
    comando1;
else
    comando2;
```

Se a condição for verdadeira, o **comando1** e o **comando2** serão executados; caso contrário, o **comando3** e o **comando4** serão executados.

```
if (condição) {
    comando1;
    comando2;
}
else {
    comando3;
    comando4;
}
```

ESTRUTURA CONDICIONAL EM C - PROBLEMA 1

Descrição do problema:

O índice de massa corpórea (IMC) de uma pessoa é igual ao seu peso (em quilogramas) dividido pelo quadrado de sua altura (em metros). A pessoa é considerada acima do peso quando seu IMC é superior a 30. Dados o peso e a altura de uma pessoa, informe se ele está acima do peso.

É fácil perceber que esse programa precisa selecionar um entre dois diagnósticos mutuamente exclusivos (normal ou acima do peso) para exibir ao usuário. Portanto, ao desenvolver seu código em linguagem C será necessário usar a **estrutura condicional composta**.

A fórmula do IMC é:

$$\text{IMC} = \frac{\text{peso em kg}}{(\text{altura em m})^2}$$

ESTRUTURA CONDICIONAL EM C - PROBLEMA 1

Solução do problema:

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

int main()
{
    float peso, altura, imc;

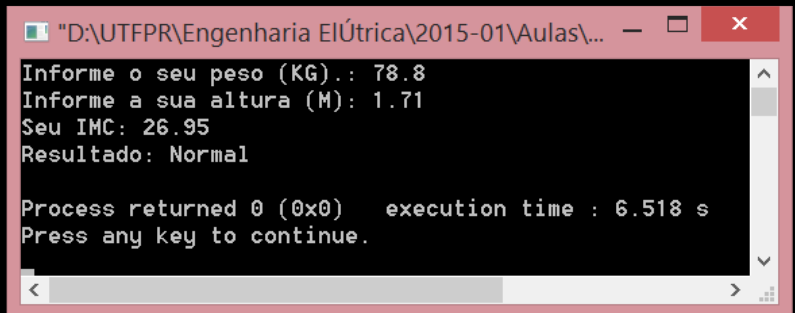
    printf("Informe o seu peso (KG).: ");
    scanf("%f%c", &peso);
    printf("Informe a sua altura (M): ");
    scanf("%f%c", &altura);

    imc = peso / pow(altura, 2);

    printf("Seu IMC: %.2f\n", imc);

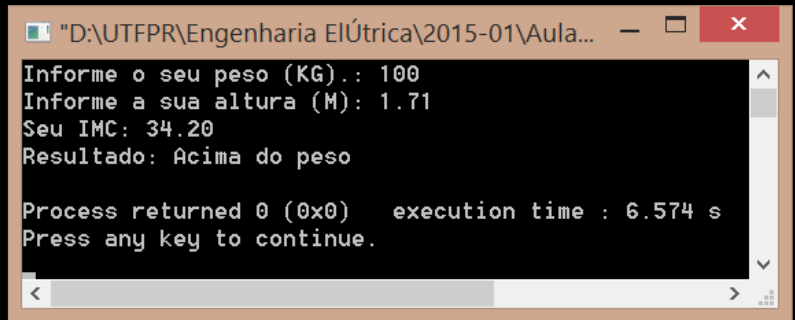
    if(imc > 30.0)
        printf("Resultado: Acima do peso\n");
    else
        printf("Resultado: Normal\n");

    return 0;
}
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aulas\... - [ ] x
Informe o seu peso (KG).: 78.8
Informe a sua altura (M): 1.71
Seu IMC: 26.95
Resultado: Normal

Process returned 0 (0x0)   execution time : 6.518 s
Press any key to continue.
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aula... - [ ] x
Informe o seu peso (KG).: 100
Informe a sua altura (M): 1.71
Seu IMC: 34.20
Resultado: Acima do peso

Process returned 0 (0x0)   execution time : 6.574 s
Press any key to continue.
```

problema_slide_12.c

ESTRUTURA CONDICIONAL EM C - PROBLEMA 2

Descrição do problema:

Em uma empresa, paga-se R\$ 19,50 por hora trabalhada e desconta-se 25% de imposto de renda dos salários superiores a R\$ 2.500,00. Dado o número de horas trabalhadas por um funcionário dessa empresa, informe o valor do desconto de IR (se houver) e o valor do salário a ser pago ao funcionário.

Esse programa precisa verificar se o salário informado é maior que dois mil e quinhentos reais para então aplicar o desconto solicitado. Portanto, ao desenvolver seu código em linguagem C será necessário usar a **estrutura condicional simples** porque você não precisa fazer nada quando o salário do funcionário for menor ou igual a R\$ 2.500,00.

Exemplo: um funcionário trabalhando 40 horas por semana totaliza no fim de um mês 160 horas trabalhadas. Então seria 160 horas * 19,50 valor da hora = R\$ 3.120,00. Logo o desconto dele será de R\$ 780,00. O salário a ser pago para este funcionário será de R\$ 2.340,00.

ESTRUTURA CONDICIONAL EM C - PROBLEMA 2

Solução do problema:

```
#include <stdio.h>
#define VALOR_HORA 19.50

int main()
{
    float horas, salario, desconto;

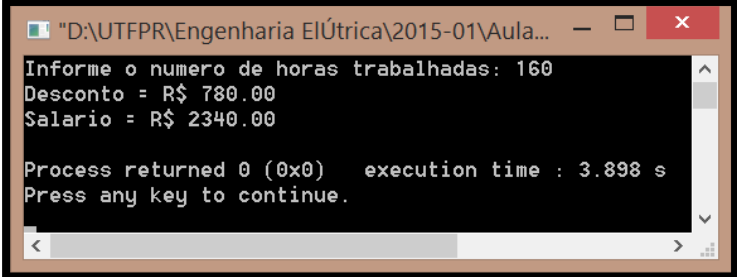
    printf("Informe o numero de horas trabalhadas: ");
    scanf("%f%c", &horas);

    salario = VALOR_HORA * horas;

    if(salario > 2500.00)
    {
        desconto = 0.25 * salario;
        printf("Desconto = R$ %.2f\n", desconto);
        salario = salario - desconto;
    }

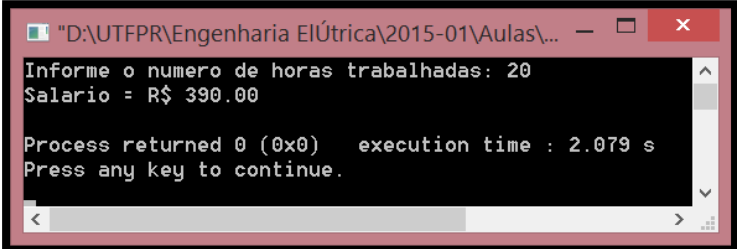
    printf("Salario = R$ %.2f\n", salario);

    return 0;
}
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aula...
Informe o numero de horas trabalhadas: 160
Desconto = R$ 780.00
Salario = R$ 2340.00

Process returned 0 (0x0)   execution time : 3.898 s
Press any key to continue.
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aulas\...
Informe o numero de horas trabalhadas: 20
Salario = R$ 390.00

Process returned 0 (0x0)   execution time : 2.079 s
Press any key to continue.
```

problema_slide_14.c

ESTRUTURA CONDICIONAL EM C - PROBLEMA 3

Descrição do problema:

Em uma universidade, a aprovação em um disciplina requer no mínimo 75% de presença e no mínimo média 6.0. Se o aluno estourar em faltas, ele é reprovado diretamente; caso contrário, se ele não alcançar a média mínima, ele fica para recuperação. Dada a média de um aluno, bem como o número de aulas da disciplina e o número de faltas do aluno, informe a sua situação.

Esse programa precisa verificar primeiro se o aluno não reprovou por faltas. Se a condição for verdadeira, é necessário verificar ainda se o mesmo atingiu a média mínima. Neste caso será usada a **estrutura condicional composta encaixada**.

Para verificar o % de faltas do aluno basta dividir o número faltas do mesmo na disciplina pelo total de aulas dadas na disciplina, multiplicar por cem e diminuir de 100% o número obtido.

ESTRUTURA CONDICIONAL EM C - PROBLEMA 3

1ª solução para o problema:

```
#include <stdio.h>

int main()
{
    float aulas_dadas, faltas, media, perc_presenca;

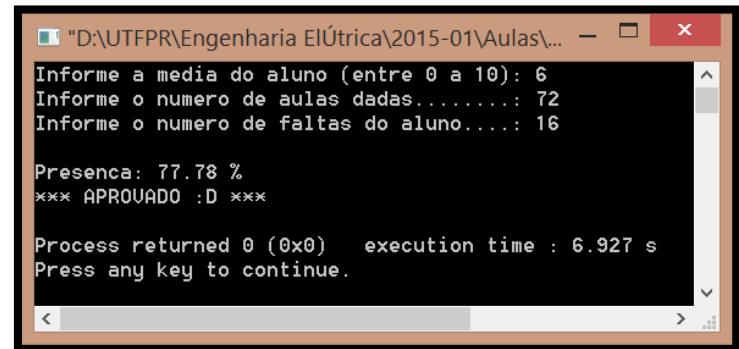
    printf("Informe a media do aluno (entre 0 a 10): ");
    scanf("%f%c", &media);
    printf("Informe o numero de aulas dadas.....: ");
    scanf("%f%c", &aulas_dadas);
    printf("Informe o numero de faltas do aluno....: ");
    scanf("%f%c", &faltas);

    perc_presenca = 100 - ((faltas / aulas_dadas) * 100);

    printf("\nPresenca: %.2f %%", perc_presenca);

    if(perc_presenca >= 75.0)
    {
        if(media >= 6.0)
            printf("\n*** APROVADO :D ***\n");
        else
            printf("\n*** RECUPERACAO :( :O ***\n");
    }
    else
        printf("\n*** REPROVADO :( ***\n");

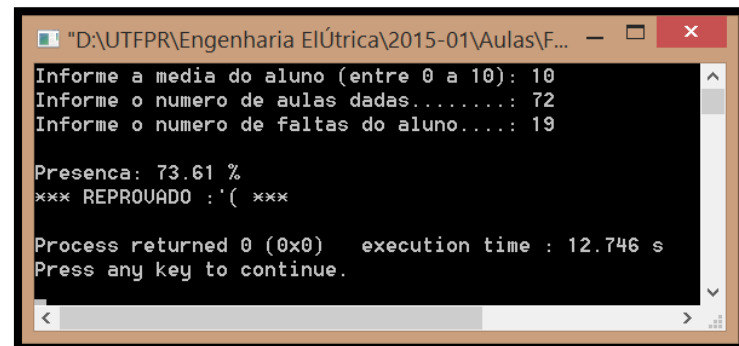
    return 0;
}
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aulas\... - [x]
Informe a media do aluno (entre 0 a 10): 6
Informe o numero de aulas dadas.....: 72
Informe o numero de faltas do aluno....: 16

Presenca: 77.78 %
*** APROVADO :D ***

Process returned 0 (0x0)   execution time : 6.927 s
Press any key to continue.
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aulas\F... - [x]
Informe a media do aluno (entre 0 a 10): 10
Informe o numero de aulas dadas.....: 72
Informe o numero de faltas do aluno....: 19

Presenca: 73.61 %
*** REPROVADO :( ***

Process returned 0 (0x0)   execution time : 12.746 s
Press any key to continue.
```

problema_slide_16_solucão1.c

ESTRUTURA CONDICIONAL EM C - PROBLEMA 3

2ª solução para o problema:

```
#include <stdio.h>

int main()
{
    float aulas_dadas, faltas, media, perc_presenca;

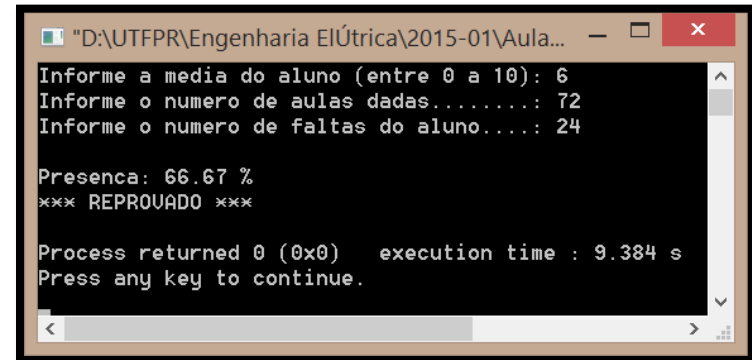
    printf("Informe a media do aluno (entre 0 a 10): ");
    scanf("%f%c", &media);
    printf("Informe o numero de aulas dadas.....: ");
    scanf("%f%c", &aulas_dadas);
    printf("Informe o numero de faltas do aluno....: ");
    scanf("%f%c", &faltas);

    perc_presenca = 100 - ((faltas / aulas_dadas) * 100);

    printf("\nPresenca: %.2f %%", perc_presenca);

    if(perc_presenca < 75.0)
        printf("\n*** REPROVADO ***\n");
    else if(media >= 6.0)
        printf("\n*** APROVADO ***\n");
    else
        printf("\n*** RECUPERACAO ***\n");

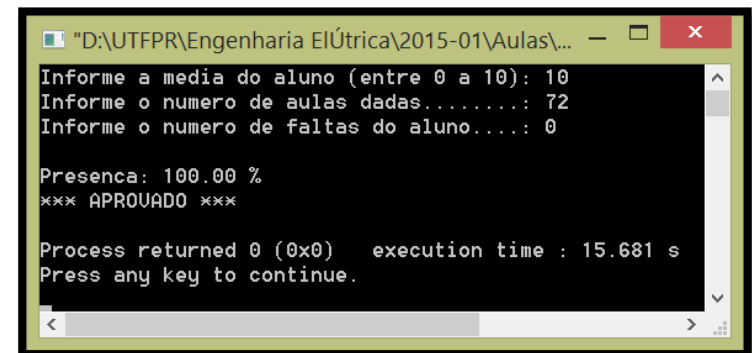
    return 0;
}
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aula...
Informe a media do aluno (entre 0 a 10): 6
Informe o numero de aulas dadas.....: 72
Informe o numero de faltas do aluno....: 24

Presenca: 66.67 %
*** REPROVADO ***

Process returned 0 (0x0)   execution time : 9.384 s
Press any key to continue.
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aulas...
Informe a media do aluno (entre 0 a 10): 10
Informe o numero de aulas dadas.....: 72
Informe o numero de faltas do aluno....: 0

Presenca: 100.00 %
*** APROVADO ***

Process returned 0 (0x0)   execution time : 15.681 s
Press any key to continue.
```

problema_slide_16_solucão2.c

ESTRUTURA CONDICIONAL EM C

Estrutura case

Em alguns programas, existem situações mutuamente exclusivas, isto é, se uma situação for executada, as demais não serão. Quando este for o caso, um comando seletivo é o mais indicado. Esse comando em C tem a seguinte sintaxe:

```
switch (variável) {  
    case valor1: lista de comandos;  
        break;  
    case valor2: lista de comandos;  
        break;  
    .....  
    default: lista de comandos;  
}
```

O comando **switch(váriável)** avalia o valor de uma variável para decidir qual **case** será executado. Cada **case** está associado a **UM** possível valor da variável, que deve ser, obrigatoriamente, do tipo **char** ou **int**. Essa estrutura é muito útil para avaliar possíveis valores de variáveis que são discretos.

ESTRUTURA SWITCH..CASE

O comando **break** deve ser utilizado para impedir a execução dos comandos definidos nos **cases** subsequentes. Quando o valor da variável não coincidir com aqueles especificados nos **cases**, será executado então o **default**. Exemplo:

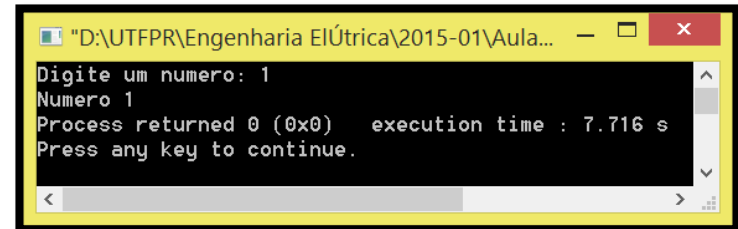
```
#include <stdio.h>

int main()
{
    int numero;

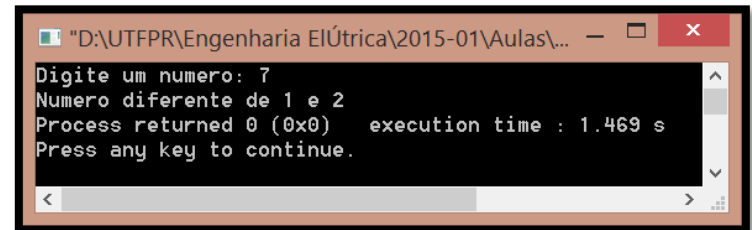
    printf("Digite um numero: ");
    scanf("%d%c", &numero);

    switch(numero)
    {
        case 1: printf("Numero 1");
               break;
        case 2: printf("Numero 2");
               break;
        default: printf("Numero diferente de 1 e 2");
    }

    return 0;
}
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aula... - [x]
Digite um numero: 1
Numero 1
Process returned 0 (0x0)   execution time : 7.716 s
Press any key to continue.
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aulas\... - [x]
Digite um numero: 7
Numero diferente de 1 e 2
Process returned 0 (0x0)   execution time : 1.469 s
Press any key to continue.
```

exemplo_slide_20.c

ESTRUTURA CONDICIONAL EM C - PROBLEMA 4

Descrição do problema:

Escreva um programa que, a partir de um mês fornecido (número inteiro de 1 a 12), apresente o nome dele por extenso ou uma mensagem de mês inválido.

Esse programa precisa verificar 13 condições (12 para cada mês e mais uma para o caso do mês informado ser inválido). A estrutura condicional **if..else** encadeada poderia ser utilizada sem problemas mas se tornaria confusa e longa. Neste tipo de problema a estrutura **switch..case** é a mais indicada.

ESTRUTURA CONDICIONAL EM C - PROBLEMA 4

Solução para o problema:

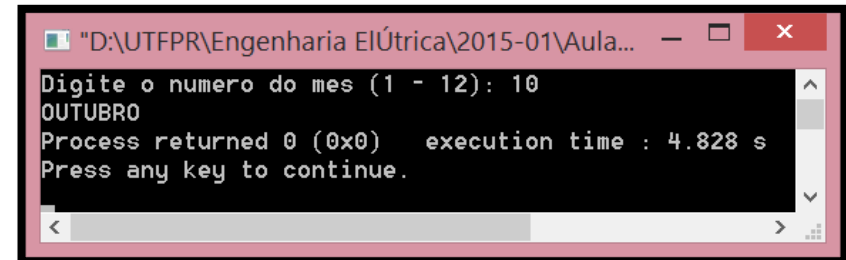
```
#include <stdio.h>

int main()
{
    int mes;

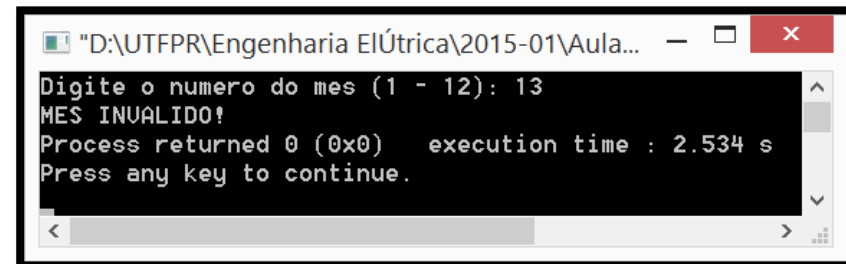
    printf("Digite o numero do mes (1 - 12): ");
    scanf("%d%c", &mes);

    switch(mes) {
        case 1 : printf("JANEIRO");      break;
        case 2 : printf("FEVEREIRO");    break;
        case 3 : printf("MARCO");        break;
        case 4 : printf("ABRIL");        break;
        case 5 : printf("MAIO");         break;
        case 6 : printf("JUNHO");        break;
        case 7 : printf("JULHO");        break;
        case 8 : printf("AGOSTO");       break;
        case 9 : printf("SETEMBRO");     break;
        case 10: printf("OUTUBRO");      break;
        case 11: printf("NOVEMBRO");     break;
        case 12: printf("DEZEMBRO");     break;
        default: printf("MES INVALIDO!");
    }

    return 0;
}
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aula... - [x]
Digite o numero do mes (1 - 12): 10
OUTUBRO
Process returned 0 (0x0)   execution time : 4.828 s
Press any key to continue.
```



```
"D:\UTFPR\Engenharia Elétrica\2015-01\Aula... - [x]
Digite o numero do mes (1 - 12): 13
MES INVALIDO!
Process returned 0 (0x0)   execution time : 2.534 s
Press any key to continue.
```

programa_slide_21.c

programa_slide_21_com_if_else.c

RESUMO

Declarações em um bloco são cercadas por { e }.

Você pode juntar condições com && e ||.

Declarações if executam o código se uma determinada condição for verdadeira.

Em C, falso é representado por 0 e verdadeiro é representado por 1.

Declarações switch verificam diversos valores de uma variável de forma eficiente.

Uma expressão condicional é uma expressão cujo valor pode ser falso ou verdadeiro.

REFERÊNCIAS BIBLIOGRÁFICAS

- ASCENCIO, A. F. G.; CAMPOS, E. A. V. D. **Fundamentos da Programação de Computadores: Algoritmos, Pascal, C/C++ (Padrão ANSI) e Java.** 3. ed. São Paulo: Pearson Education do Brasil, 2012. 569 p.
- FORBELLONE, A. L. V.; EBERSPACHER, H. F. **Lógica de Programação: A construção de algoritmos e estruturas de dados.** 3. ed. São Paulo: Prentice Hall, 2005. 218p.
- PEREIRA, S. D. L. **Algoritmos e Lógica de Programação em C: Uma abordagem didática.** 1. ed. São Paulo: Érica, 2010. 190 p.