

**UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO
CIC107 – PROGRAMAÇÃO DE COMPUTADORES I**

7ª Aula prática

Estrutura condicional - SWITCH

O *switch* é um comando seletivo usado quando é preciso selecionar uma de várias opções em um programa. Cada opção está associada a um comando (ou uma lista de comandos). Quando um comando for executado, os demais não serão (são mutuamente exclusivos).

O *switch* calcula uma *expressão* e compara o valor resultante com diversas constantes. O fluxo do programa é desviado para a constante que for igual ao valor da expressão. Na linguagem C/C++ essa estrutura tem a seguinte sintaxe:

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

- ***switch(expressão)***: avalia o valor de uma expressão para decidir qual *case* será executado. A expressão deve ser de um tipo ordinal (*int*, *char*, *short*, *long*, etc). *Float* e *double* não são permitidos.
- ***case***: cada *case* está associado a apenas um possível valor da expressão.
- ***break***: esse comando deve ser utilizado para impedir que sejam executados os comandos definidos nos *cases* seguintes, pois o *switch* executa seu bloco em cascata. Ou seja, se a variável indicar para o primeiro *case* e dentro do *switch* tiver outros *cases*, o *switch* executará todos os outros *case* a não ser que utilizemos o comando *break* para sair do *switch*.
- ***default***: quando o valor da variável não coincidir com aqueles especificados nos *case*, será executada então a lista de comandos da opção *default*. Seu uso é opcional e não precisa existir no *switch*. Se o *default* não existir e a expressão não for igual a nenhuma das constantes especificadas, então nada será executado.

O comando *switch* também é um comando condicional como o *if/else*. A diferença é que os *case* não aceitam operadores lógicos. Portanto, não é possível fazer uma comparação. Isso limita o *case* a apenas valores definidos.

Considere o seguinte exemplo: programa que lê o conceito obtido pelo aluno no semestre (de A a F) e imprime seu significado.

```
#include <iostream>  
using namespace std;  
main(){  
  char Conceito;  
  cout << "Digite o conceito obtido no semestre:" << endl;  
  cin >> Conceito;  
  switch( Conceito ){  
    case 'A' : cout << "Excelente"; break;
```

```

    case 'B' : cout << "Otimo"; break;
    case 'C' : cout << "Bom"; break;
    case 'D' : cout << "Regular"; break;
    case 'F' : cout << "Ruim"; break;
    default : cout << "O conceito informado nao existe.";
}

cout << endl;
system("pause");
}

```

Como neste exemplo, a situação mais comum é que a *expressão* do comando *switch* seja constituída apenas por uma variável.

Observe o mesmo programa, porém com a utilização de *if/else*. Neste programa, a construção é mais longa e pode facilmente levar a erros:

```

#include <iostream>
using namespace std;

main(){
    char Conceito;
    cout << "Digite o conceito obtido no semestre:" << endl;
    cin >> Conceito;

    if ( Conceito == 'A')
        cout << "Excelente";
    else if ( Conceito == 'B')
        cout << "Otimo";
    else if ( Conceito == 'C')
        cout << "Bom";
    else if ( Conceito == 'D')
        cout << "Regular";
    else if ( Conceito == 'F')
        cout << "Ruim";
    else cout << "O conceito informado nao existe.";

    cout << endl;
    system("pause");
}

```

Exercícios:

- 1) Escreva um programa que peça para o usuário entrar um número correspondente a um dos meses do ano e que então apresente na tela o nome do mês e o número de dias desse mês. Utilize o comando *switch*. Se o número digitado não corresponder a um mês válido no calendário, escrever a mensagem 'Mês Inválido'.
- 2) Calculadora de quatro operações usando *switch*: faça um programa que, dados dois números e um caracter (+, -, *, /), calcule e imprima o resultado da aplicação, sobre esses números, da operação indicada pelo caracter.
Se a opção digitada pelo usuário for inválida, mostrar uma mensagem de erro. Lembre-se que na operação de divisão o segundo número deve ser diferente de zero. Como garantir isso?
- 3) Faça um programa que realize conversões de temperaturas em graus Celsius, Fahrenheit ou Kelvin. O programa deve ler da entrada a temperatura e uma das seguintes opções:
 - A. de Celsius para Fahrenheit
 - B. de Fahrenheit para Celsius
 - C. de Celsius para Kelvin

- D. de Kelvin para Celsius
- E. de Kelvin para Fahrenheit
- F. de Fahrenheit para Kelvin

Utilize o comando *switch*. Lembre-se que:

Conversões		
de	para	fórmula
Celsius	Fahrenheit	$^{\circ}\text{F} = ^{\circ}\text{C} \times 1,8 + 32$
Fahrenheit	Celsius	$^{\circ}\text{C} = (^{\circ}\text{F} - 32) \times 5/9$
Celsius	Kelvin	$\text{K} = ^{\circ}\text{C} + 273,15$
Kelvin	Celsius	$^{\circ}\text{C} = \text{K} - 273,15$
Kelvin	Fahrenheit	$^{\circ}\text{F} = \text{K} \times 1.8 - 459.67$
Fahrenheit	Kelvin	$\text{K} = (^{\circ}\text{F} + 459.67) / 1.8$

- 4) Para se determinar o número de lâmpadas necessárias para cada cômodo de uma residência, existem normas que dão o mínimo de potência de iluminação exigida por metro quadrado (m^2), conforme a utilização deste cômodo.

Seja o seguinte quadro tomado como exemplo:

UTILIZAÇÃO	CLASSE	POTÊNCIA/ m^2
quarto	1	15
sala	1	15
cozinha	2	18
varanda	2	18
escritório	3	20
banheiro	3	20

Supondo-se que só serão utilizadas lâmpadas de 60 w, fazer um programa que:

- a) leia um número indeterminado de linhas contendo cada uma:
 - a classe de um cômodo de uma residência;
 - as dimensões do cômodo (considere apenas cômodos retangulares) ;
 A última linha terá classe igual a zero e dimensões também nulas, para indicar fim dos dados.
 - b) calcule para cada cômodo: a sua área, a potência de iluminação e o número de lâmpadas necessárias;
 - c) calcule para toda a residência: o total de potência e o total de lâmpadas.
 - d) imprima o número de lâmpadas total. Se esse número for fracionário, aproximar para o inteiro imediatamente superior. Exemplo: 8,3 \rightarrow 9.
- Obs.: para converter uma variável x do tipo float ou double em inteiro, pode-se usar:
 $k = (\text{int}) x$;
 onde k deve ser uma variável inteira. O valor atribuído a k será obtido pelo truncamento do valor de x .