



LÓGICA DE PROGRAMAÇÃO

PROF^a. M.Sc. JULIANA H Q BENACCHIO



- Também chamado de estrutura de seleção múltipla (caso).
- Assim como o **if-else**, o **switch** também é uma estrutura de seleção.
- O **if-else** realiza o teste da condição e considera apenas duas alternativas para a expressão: verdadeiro ou falso.
- Já o **switch** é próprio para se testar uma variável em relação a **diversos valores pré-estabelecidos**

Estrutura Condicional: **switch-case**



```
switch (variável)
{
    case constante1:
        sequencia de comandos;
        break;
    case constante2:
        sequencia de comandos;
        break;
    .
    .
    .
    case constante_n:
        sequencia de comandos;
        break;
    default:
        sequencia de comandos;
}
```



- O **switch** testa a variável entre parênteses e compara seu valor com os rótulos dos casos (**case**). A variável só pode ser do tipo **int** ou **char**, não aceita outros tipos de variáveis, como por exemplo, o tipo **float**.
- Cada **case** deve ser rotulado por uma constante, também do tipo inteiro ou caractere. Esta constante deve ser terminada por dois pontos (:). Os rótulos dos casos devem ser todos diferentes.



- Pode haver um ou mais comandos dentro de cada case. Essa sequência de comandos não necessita estar entre chaves. Apenas o comando principal **switch** deve possuir as chaves de início e fim.
- O **switch** testa a variável e executa a sequência de comandos cujo case corresponda ao valor atual da variável. A sequência de comandos será executada até que o comando **break** seja alcançado.



- O comando **break** é um dos comandos de desvio em C.
- Quando um **break** é encontrado em um **switch**, causa uma saída imediata e a execução do programa “salta” para a linha de código seguinte ao comando **switch**.



- Se nenhum case for satisfeito, a sequência de comandos existentes em **default** é executada.
- O **default** é opcional e, se não estiver presente, nenhum comando será executado se todos os testes falharem.
- O **default** de um **switch** não necessita de **break**, porque depois de executar as instruções associadas ao último comando, termina a execução do **switch**.

Estrutura Condicional: `switch-case`



```
int num;
printf ("Digite um numero: ");
scanf ("%d",&num);
switch (num)
{
    case 9:
        printf ("\n\n0 numero e igual a 9.\n");
        break;
    case 10:
        printf ("\n\n0 numero e igual a 10.\n");
        break;
    case 11:
        printf ("\n\n0 numero e igual a 11.\n");
        break;
    default:
        printf ("\n\n0 numero nao e nem 9 nem 10 nem 11.\n");
}
```




- Outra diferença entre o **if-else** é que a estrutura **switch** não aceita expressões: aceita apenas constantes.

```
int num;
...
if (num % 2 == 0)
    printf("PAR");
else
    printf("IMPAR");
```

```
int num, res;
...
res = num % 2;
switch (res)
{
    case 0:
        printf("PAR");
        break;
    default:
        printf("IMPAR");
}
```



- Mas é importante perceber qual comando devemos utilizar em cada situação!
- Por exemplo, no problema para verificar se uma idade é maior ou igual a 18, o comando **switch** não pode ser utilizado, uma vez que o **switch** só pode testar **igualdade**, diferente do **if-else** que pode avaliar uma expressão lógica ou relacional.

Estrutura Condicional: `switch-case`



```
int idade;  
  
printf("Digite a idade: ");  
scanf("%d", &idade);  
  
if (idade >= 18)  
    printf("Maior de idade\n");  
  
else  
    printf("Menor de idade\n");
```

- Nesse caso, é impossível utilizar o comando **`switch`**.



- No exemplo abaixo é possível verificar que a utilização do comando **switch** é melhor aplicado do que a utilização do comando **if-else**.

```
char turno;
...
if ( turno == 'M' )
    printf("Matutino");
else
    if (turno == 'V' )
        printf("Vespertino");
    else
        if (turno == 'N' )
            printf("Noturno");
        else
            printf("Turno invalido");
```

**Muitos if-else
aninhados!!**

Estrutura Condicional: `switch-case`



```
char turno;
...
switch (turno)
{
    case 'M':
        printf("Matutino");
        break;
    case 'V':
        printf("Vespertino");
        break;
    case 'N':
        printf("Noturno");
        break;
    default:
        printf("Turno invalido");
}
```

**Melhor utilizar
o switch!!**

Estrutura Condicional: `switch-case`



- Exemplo para verificar o sexo:

```
char sexo;
...
switch (sexo)
{
    case 'm':
        printf("Sexo masculino");
        break;
    case 'M':
        printf("Sexo masculino");
        break;
    case 'f':
        printf("Sexo feminino");
        break;
    case 'F':
        printf("Sexo feminino");
        break;
    default:
        printf("Sexo digitado incorretamente");
}
}
```



“Passar caindo” pelos cases:

- Tecnicamente, os comandos **break**, dentro do **switch**, são opcionais. Eles terminam a seqüência de comandos associados com cada constante. Se o comando **break** é omitido, a execução continua pelos próximos cases até que um **break**, ou o fim do **switch**, seja encontrado.

Estrutura Condicional: **switch-case**



- Alterando o exemplo para verificar o sexo:

```
char sexo;
...
switch (sexo)
{
    case 'm':
    case 'M':
        printf("Sexo masculino");
        break;
    case 'f':
    case 'F':
        printf("Sexo feminino");
        break;
    default:
        printf("Sexo digitado incorretamente");
}
```


“Passar caindo” pelos case's

Com esse exemplo é possível verificar duas situações no comando **switch**:

1. Você pode ter cases sem comandos associados
2. A execução continua no próximo **case** se nenhum comando **break** estiver presente

Quando isso ocorre, a execução simplesmente “cai” no **case** seguinte

O fato de os **case**'s poderem ser executados em conjunto quando nenhum **break** estiver presente evita a duplicação indesejável de código, resultando em um código mais eficiente.



Utilização como menu:

- O comando **switch** é frequentemente usado para processar uma entrada, via teclado, como em uma seleção por menu.
- Por exemplo, o programa para simular uma calculadora:

Estrutura Condicional: switch-case



```
int main( )
{
    int op;
    float num1, num2, res;

    printf("Calculadora\n\n");
    printf("Digite os numeros para realizar a operacao:\n");
    printf("Primeiro numero:");
    scanf("%f", &num1);
    printf("Segundo numero:");
    scanf("%f", &num2);

    printf("\nEscolha a opcao:\n");
    printf("1. Soma\n");
    printf("2. Subtracao\n");
    printf("3. Multiplicacao\n");
    printf("4. Divisao\n");
    scanf ("%d", &op);
```

Estrutura Condicional: `switch-case`



```
switch (op)
{
    case 1:
        res = num1 + num2;
        printf("\nResultado da soma = %.2f\n\n", res);
        break;
    case 2:
        res = num1 - num2;
        printf("\nResultado da subtracao = %.2f\n\n", res);
        break;
    case 3:
        res = num1 * num2;
        printf("\nResultado da multiplicacao = %.2f\n\n", res);
        break;
}
```

Estrutura Condicional: `switch-case`



```
    case 4:
        if (num2 == 0)
            printf ("\nDivisao por zero!\n\n");
        else
        {
            res = num1 / num2;
            printf ("\nResultado da divisao = %.2f\n\n", res);
        }
        break;
    default:
        printf ("\nOpcao inexistente!\n");
}
return 0;
}
```