

# Aula 02 - Introdução à Linguagem C

## Programação: Estrutura-Seleção

### OPL e IB

Prof: Anilton Joaquim da Silva

[Anilton.ufu@outlook.com](mailto:Anilton.ufu@outlook.com)

# Linguagem C:

## Estrutura de Seleção

- Até agora não aprendemos a dizer para o computador “Se for assim, então faça assado! Senão, faça cozido!”
- Como exemplo no calculo das raizes reais do segundo grau temos que verificar o valor calculado do DELTA:
  - Se  $\text{DELTA} < \text{ZERO}$ , então não existe raizes reais par serem calculadas
  - Se  $\text{DELTA} == \text{ZERO}$ , então as raizes são iguais
  - Se  $\text{DELTA} > \text{ZERO}$ , então existem duas raizes reais e as mesmas devem ser calculadas
- As instruções de seleção utilizam dos operadores **relacionais** e **lógicos** para definição de qual caminho deve ser seguido na execução.

# Linguagem C:

## Operadores/Expressões

- Relacionais

- `==` igual a
- `!=` diferente de
- `>` maior que
- `<` menor que
- `>=` maior ou igual a
- `<=` menor ou igual a

- Lógicos

- `&&` and
- `||` ou
- `!` not

A	!A
V	F
F	V

A	B	A && B
V	V	V
V	F	F
F	V	F
F	F	F

A	B	A    B
V	V	V
V	F	V
F	V	V
F	F	F

# Linguagem C:

## Comando - if-else

- Em C, testes simples (tudo o que você realmente precisa) podem ser feitos com a estrutura if, que tem uma das seguintes sintaxes:

- **if (expressão lógica)**  
**bloco de comandos 1**

- **if (expressão lógica)**  
**bloco de comandos 1** (se exp lógica Verdadeiro executa)

**else**

**bloco de comandos 2** (se exp lógica é Falso executa)

# Linguagem C:

## Comando - if-else

- Uma expressão lógica é uma expressão cuja avaliação resulte em **verdadeiro** ou **falso** como, por exemplo, as expressões que usam os operadores relacionais
- Um bloco de comandos é ou uma instrução ou um conjunto de instruções dentro de { }.
- Quando a expressão lógica é avaliada, se seu resultado for verdadeiro, então o bloco de comandos 1 será executado, e ignora o bloco de comando 2. Se o resultado for falso, o bloco de comandos 2 será executado, e ignora o bloco de comando 1. E segue a execução na próxima linha de comando.

**OBS:** Se no comando de decisão não existir o **else** o programa continua execução na próxima linha de comando.

# Linguagem C:

## Exemplo - Raizes de Eq de 2º grau

```
• #include <stdio.h>
• #include <math.h> // funções matemáticas

• int main()
• {
•     float a, b, c, d, x1, x2;

•     printf("Calculo raizes da eq do 2o grau - AX2 + BX + C = o. \n");
•     // exemplo: X^2 - 5*X + 6 = 0   x1 = 3   x2 = 2

•     printf("Digite o valor de a: ");
•     scanf("%f", &a);
•     printf("Digite o valor de b: ");
•     scanf("%f", &b);
•     printf("Digite o valor de c: ");
•     scanf("%f", &c);
•     d = pow( b, 2) - 4*a*c;
•     printf("\nDelta: %f", d);

•     if (d < 0)
•     {
•         printf("\nA equação nao tem raizes reais");
•     }

•     else
•     {
•         if ( d == 0)
•         {
•             x1 = -b/(2*a);
•             printf ("\nA equacao tem duas raizes iguais! x1 e x2 = %f ", x1);
•         }
•         else //continua a direita
•         {
•             printf("\nA equacao tem duas raizes distintas! X1 e x2 ");
•             x1 = (-b + sqrt(d))/(2*a);
•             x2 = (-b - sqrt(d))/(2*a);
•             printf("\nX1 = %f ", x1);
•             printf("\nX2 = %f ", x2);
•         } // fim do else, do 2.o if
•     } // fim do else, do 1.o if

•     printf("\n");
•     return 0;
• } // fim da função main - do programa
```

# Linguagem C:

## Prioridade de operadores

- Para operadores lógicos:
  - – primeiro o NOT (!), depois o AND (&&) e por último o OR (||)
- Expressões com vários tipos de operadores:
  - 1. Parênteses, incremento e decremento (++ , --)
  - 2. not (!)
  - 3. Multiplicação, divisão e módulo (o que aparecer primeiro);
  - 4. Soma e subtração;
  - 5. Operadores relacionais (<, <=, >, >=)
  - 6. Operadores relacionais (==, !=)
  - 7. Operadores Lógicos: and (&&)
  - 8. Operadores Lógicos: or (||)
  - 9. Operadores de Atribuição (=, +=, -=, \*=, /=, %=)

# Linguagem C:

## Comando - switch

- Seja um programa que leia os valores numéricos de uma data: dia/mês/ano, e imprima a data com o nome do mês correspondente:
- ...
- `int mes;`
- ...
- `printf("Digite o numero do mes: " );`
- `scanf("%d", &mes);`
- ...
- `if (mes == 1)`
- `printf("Janeiro");`
- `if (mes == 2)`
- `printf("Fevereiro");`
- ...
- `if (mes == 12)`
- `printf("Dezembro");`
- ...



# Linguagem C:

## Comando - **switch**

- Em vez de usar vários if e else-if, uma solução melhor seria usar switch, criado exatamente para tratar estas situações. A sintaxe do uso do switch é a seguinte:
  - switch (identificador)
  - {
    - case valor1: bloco\_comandos1
    - case valor2: bloco\_comandos2
    - ...
    - case valorN: bloco\_comandosN
    - default: bloco\_comandos\_default
  - }

# Linguagem C:

## Exemplo - comando **switch**

```
• /* Seja um programa que leia os valores numéricos de uma data:
• dia/mês/ano, e imprima a data com o nome do mês correspondente:*/

• #include <stdio.h>
• #include <stdlib.h>

• int main()
• {
•     int dia, mes, ano;

•     printf("Entre com a data: \n");
•     printf("Dia? ");
•     scanf("%d", &dia);
•     printf("Mes? ");
•     scanf("%d", &mes);
•     printf("Ano? ");
•     scanf("%d", &ano);
•     printf("\n A data eh: %d de ", dia);
•     switch (mes)
•     {
•     case 1:
•         printf("Janeiro");
•         break;
•     case 2:
•         printf("Fevereiro");
•         break;
•     case 3:
•         printf("Marco");
•         break;
•     case 4:
•         printf("Abril");
•         break; //continua a direita

•     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("Erro-mes inexistente!");
•     }
•     printf(" de %d \n", ano);

•     printf("\n");
•     return 0;
• } // fim da função main - do programa
```

# Linguagem C:

## Exemplo - comando switch

- Implemente um programa que:
- Leia três números reais quaisquer;
- Imprima o seguinte menu na tela:
  - - 1. Soma
  - - 2. Média
  - - 3. Menor
  - - 4. Maior
- Leia o valor selecionado, o número da opção escolhida;
- Para decidir o que fazer, o número da opção escolhida, deve ser testado com o switch. Cada opção deve, respectivamente, permitir que calcule e imprima o que se pede.

# Linguagem C:

## Exemplo - comando switch

```
• #include <stdio.h>
• #include <stdlib.h>

• int main()
• {
•     float a, b, c, s, x, y;
•     int opcao;

•     printf("Digite tres numero inteiros: \n");
•     printf("Numero 1: ");
•     scanf("%f", &a);
•     printf("Numero 2: ");
•     scanf("%f", &b);
•     printf("Numero 3: ");
•     scanf("%f", &c);
•     printf("\nEscolha uma das opções abaixo: \n");
•     printf("1 - Soma \n2 - Media \n3 - Menor \n4 - Maior\n      ? : ");
•     scanf("%d", &opcao);

•     switch(opcao)
•     {
•         case 1:
•             s = a + b + c;
•             printf("A soma dos tres numeros eh %f", s);
•             break;
•         case 2:
•             printf("A media dos tres numeros eh %f", (a + b + c)/3);
•             break; // continua a direita

•         case 3:
•             if((a < b) && (a < c))
•             {
•                 x = a;
•             }
•             else
•             {
•                 if(b < c)
•                 {
•                     x = b;
•                 }
•                 else
•                 {
•                     x = c;
•                 }
•             } // fim do else do 1.0 if
•             printf("O menor dentre os tres numeros eh %f", x);
•             break;
•         case 4:
•             if((a > b) && (a > c)){y = a;}
•             else if(b > c){y = b;}
•             else {y = c;}
•             printf("O maior dentre os tres numeros eh %f", y);
•             break;
•         default:
•             printf("Opcao invalida. Execute o programa novamente e leia direito as opcoes.");
•     }

•     printf("\n");
•     return 0;
• }
```