

Introducao a Programacao

Estruturas de decisao

`periclesmiranda@gmail.com`

Plano de aula

- 1 Estrutura `if-then`
- 2 Estrutura `if-then-else`
- 3 O operador condicional ternário
- 4 Estrutura `switch`

Estruturas de decisão

Permite

- Execução de instruções condicionada ao fato de que uma condição seja satisfeita
- Executar ações alternativas

Exemplos

- Se tiver dinheiro suficiente, irei ao cinema
- Se $y \neq 0$, divida x por y
- Se tiver bolo de chocolate, comerei bolo de chocolate
senão comerei torta

- 1 Estrutura if-then
- 2 Estrutura if-then-else
- 3 O operador condicional ternário
- 4 Estrutura switch

Estruturas de decisão

if-then

- Execução de instruções condicionada ao fato de que uma expressão lógica seja satisfeita.
- Expressão lógica retorna algum valor que pode ser
0 : False
1 : True
- Tipicamente, usa operadores relacionais:
($i \neq 0$) retorna 1 se i é diferente de 0,
senão retorna 0

if-then

Sintaxe

```
if (expressao logica)
{
    Sequencia de instrucoes
}
```

Exemplo

```
if (x > y)
{
    printf("%f e maior que %f\n", x, y);
}
```

Sintaxe

Observações

- Não tem ; depois do `if` (expressao)
- As () são necessárias ao redor da expressão
- As chaves { } delimitam a sequência de instruções
- Toda chave { tem que ser fechada por uma chave }
- Sequência de instruções: pode conter outros `if-then`
- Chaves { } opcionais para apenas uma instrução
- Indentar o código!

Exemplo

```
if (x > y)
    printf("%f e maior que %f\n", x, y);
```

Exercício

Programa divisão

Escrever um programa que peça ao usuário para entrar com dois números de ponto flutuante x e y , e que divida x por y se $y \neq 0$.

Exercício

Programa maior

Escrever um programa que peça ao usuário para entrar com dois números de ponto flutuante x e y , e que imprima o maior dos dois na tela.

- 1 Estrutura `if-then`
- 2 Estrutura `if-then-else`
- 3 O operador condicional ternário
- 4 Estrutura `switch`

Estruturas de decisão

if-then-else

- Decide entre duas sequências de comandos qual das duas vai ser executada
- Uma e somente uma das duas sequências vai ser executada!

Sintaxe

```
if (expressao logica)
{
    sequencia de instrucoes 1
}
else
{
    sequencia de instrucoes 2
}
```

if-then-else

Exemplo

```
if (i % 2 == 1)
{
    printf("%d e um numero impar\n", i);
}
else
{
    printf("%d e um numero par\n", i);
}
```

Exercício

Programa triângulo

Escrever um programa que peça ao usuário para entrar com três valores, e que verifica se é possível gerar um triângulo com estes tamanhos de lado.

(Verificar a desigualdade triangular: $a \leq b + c$)

Exercício

Programa tênis

Quero comprar um par de tênis para correr. O modelo comum custa R\$300 e o modelo de competição custa R\$600. Escreva um programa que pergunte quanto de dinheiro tenho (valor inteiro), e que diga se posso comprar um par de tênis, e se sim, qual modelo seria.

Observação

Cuidado

Sem as chaves { }, o else é relacionado com o último if

```
if (expressao1)
    if (expressao2)
        instrucao1;
    else
        instrucao2;
```

```
if (expressao1)
{
    if (expressao2)
        instrucao1;
}
else
    instrucao2;
```

- 1 Estrutura `if-then`
- 2 Estrutura `if-then-else`
- 3 O operador condicional ternário
- 4 Estrutura `switch`

Estruturas de decisão

O operador condicional ternário

Uso: Quando as duas opções de um `if-then-else` contém apenas uma atribuição a uma mesma variável

```
if (expressao logica)
{
    variavel = expressao1;
}
else
{
    variavel = expressao2;
}
```

O operador condicional ternário

Sintaxe

```
variavel = (Expr. logica) ? Expr. 1 : Expr. 2;
```

Exemplo

```
maior = (x > y) ? x : y;
```

Exercício

Programa negativo

Escrever um programa que peça ao usuário um valor de ponto flutuante, e que atribua a uma variável:

- a raiz quadrada dele se ele for positivo;
- o quadrado dele se ele for negativo.

- 1 Estrutura `if-then`
- 2 Estrutura `if-then-else`
- 3 O operador condicional ternário
- 4 Estrutura `switch`

Estrutura switch

Permite

Várias decisões possíveis dependendo de uma variável/expressão

Sintaxe

```
switch (expressao)
{
    case 1: { sequencia de comandos 1; break; }
    case 2: { sequencia de comandos 2; break; }
    ...
    default: { sequencia de comandos n; }
}
```

switch

Observações

- expressão: retorna um inteiro `int` ou um carácter `char`
- Executa sequência de comandos do `case` correspondendo ao valor da expressão...
...e as sequências seguintes se não tiver `break`
- `default` (opcional): valores não mencionados nos `case`

Estrutura switch

Exemplo: programa caixa eletrônica

Escrever um programa que gerencia um caixa eletrônico de banco. O usuário entra com um inteiro que representa uma opção possível:

- 1 Saldo em conta corrente
- 2 Extrato bancário
- 3 Sacar dinheiro
- 4 Transferência
- 5 ...

Exemplo: programa caixa eletrônica

Possível implementação (usando if-then-else)

```
if (i == 1)
{
    printf("Saldo\n");
}
else
    if (i == 2)
    {
        printf("Extrato\n");
    }
    else
        if (i == 3)
        {
            printf("Saque\n");
        }
        else
            if (i == 4)
            {
                printf("Transferencia\n");
            }
            else
            {
                printf("Opcao invalida\n");
            }
        }
```


Exemplo: programa caixa eletrônica

Possível implementação (usando switch)

```
switch(i)
{
    case 1:
        printf("Saldo\n");
        break;
    case 2:
        printf("Extrato\n");
        break;
    case 3:
        printf("Saque\n");
        break;
    case 4:
        printf("Transferencia\n");
        break;
    default:
        printf("Opcao invalida\n");
}
```

switch

Exemplo 2: Influência do break e do default

```
int i;
scanf("%d", &i);
switch(i)
{
    case 1:
        printf("Voce digitou %d\n", i);
    case 2:
        printf("Voce digitou %d\n", i);
    case 3:
        printf("Voce digitou %d\n", i);
    default:
        printf("Voce digitou nem 1, nem 2, nem 3\n");
}
```

Exercício

Programa calculadora

Escrever um programa que implementa uma calculadora de 4 operações(+, -, *, /), usando o `switch` para realizar a operação escolhida pelo usuário.

Exemplo

O usuário entra:

4 * 8

O programa retorna 32