

Algoritmia e Programação

Controlo de fluxo: estruturas de decisão

Tratamento consoante o caso...

- A forma como resolvemos um problema pode depender dos próprios dados do problema.
 - Por exemplo, no controlo de temperatura de uma sala podemos...
 - escolher *aquecer* se estiver *frio*, ou
 - escolher *arrefecer* se estiver *calor*, ou
 - simplesmente *ventilar* se estiver uma *temperatura confortável*.

Decidir o que fazer...

- Diferentes situações podem ditar executar diferentes passos para resolver uma instância do problema.
- Um algoritmo deverá permitir exprimir que acções deverão ser executadas consoante uma **condição**.
- A avaliação da condição só pode resultar em dois valores possíveis:
 - **VERDADEIRO**
 - **FALSO**

Declaração condicional

Declaração condicional

- A declaração condicional permite executar um bloco de código **somente** se uma condição lógica se verificar...
- ... mas também permitem executar um bloco de código alternativo (e opcional), caso a condição lógica não se verifique.
- Ou seja, permite executar exclusivamente um de dois blocos de código, dependendo do valor lógico de uma condição.

Exemplo trivial

- No quotidiano, podemos escolher o calçado dependendo das condições meteorológicas.
 - “**Se** *estiver a chover* **então** *levo botas*, **senão** *levo sapatilhas*.”

Exemplo trivial

- No quotidiano, podemos escolher o calçado dependendo das condições meteorológicas.
- “**Se** *estiver a chover* **então** *levo botas*, **senão** *levo sapatilhas*.”

Acção
condicional
(negativa)

Condição
lógica

Acção
condicional
(afirmativa)

Formalizando...

- Esta construção é utilizada nos algoritmos para exprimir uma declaração condicional:

```
SE (condição lógica)  
ENTÃO  
    bloco afirmativo  
SENÃO  
    bloco negativo  
FIM SE
```

- O bloco negativo é **opcional**: pode não ter **SENÃO**.

Caso prático

- Indique se o valor da variável x é par ou ímpar.

```
SE ( $x \text{ MOD } 2 = 0$ )  
ENTÃO  
    ESCREVER("x é par.")  
SENÃO  
    ESCREVER("x é ímpar.")  
FIM SE
```

- Nota: o operador MOD devolve o resto da divisão inteira.

Composição de declarações condicionais

- Podemos compor declarações condicionais para obter execução diferenciada para mais do que dois blocos exclusivos.
 - Dentro de uma acção condicional, insere-se uma nova declaração condicional.

Exemplo

- Suponha que o comportamento dinâmico de um drone depende do seu peso, que varia com a carga que transporta
 - é leve quando o seu peso bruto é inferior a 20 kg;
 - é pesado quando o seu peso bruto é superior a 50 kg;
 - é ideal quando o seu peso bruto se situa entre os 20 e os 50 kg.

Exemplo

- Suponha que o comportamento dinâmico de um drone depende do seu peso, que varia com a carga que transporta
 - é leve quando o seu peso bruto é inferior a 20 kg;
 - é pesado quando o seu peso bruto é superior a 50 kg;
 - é ideal quando o seu peso bruto se situa entre os 20 e os 50 kg.

```
SE (peso_bruto < 20)
ENTÃO
    ESCREVER("Leve!")
SENÃO
    SE (peso_bruto <= 50)
    ENTÃO
        ESCREVER("Ideal.")
    SENÃO
        ESCREVER("Pesado!")
    FIM SE
FIM SE
```

Declaração de comutação

Declaração de comutação

- Utiliza-se quando a acção a realizar depende de uma variável ou uma expressão que pode tomar um conjunto limitado de valores discretos.

```
CASO opcao SEJA
'A', 'E':
    uma_variavel_qualquer = 5
'I':
    uma_variavel_qualquer = 1
    ESCRIVER("Voltou ao início.")
OUTROS:
    ESCRIVER("Nenhuma operação.")
FIM CASO
```

Implementação em C

Declaração condicional

- Em C, utiliza-se a construção *if-else*.

```
if(condição)  
{  
    // bloco afirmativo  
}  
else  
{  
    // bloco negativo  
}
```

- O **else** é **opcional**: pode não ser necessário!

Declaração condicional

```
int x;  
// ...  
printf("x = ");  
scanf("%d", &x);  
  
if((x % 2) == 0)  
{  
    printf("%d é par.\n", x);  
}  
else  
{  
    printf("%d é ímpar.\n", x);  
}
```

Declaração condicional

```
int peso_bruto;  
// ...  
printf("Peso bruto: ");  
scanf("%d", &peso_bruto);  
  
if(peso_bruto < 20)  
{  
    printf("Leve!!!\n");  
}  
else if(peso_bruto <= 50)  
{  
    printf("Ideal.\n");  
}  
else  
{  
    printf("Pesado!!!!\n");  
}
```

Declaração de comutação

- Em C, utiliza-se a construção *switch-case-default*.

```
switch(valor)
{
    case valor1:
        // Trata caso 1.
        break; // Se não colocar break, continua nas linhas seguintes.
    case valor2: case valor3:
        // Trata casos 2 e 3.
        break;
    default:
        // Trata todos os outros casos.
        break;
}
```

Declaração condicional

```
char opcao;  
int uma_variavel_qualquer;  
// ...  
printf("Opção = ");  
scanf("%c", &opcao);  
  
switch(opcao)  
{  
case 'A': case 'E':  
    uma_variavel_qualquer = 5;  
    break;  
case 'I':  
    uma_variavel_qualquer = 1  
    printf("Voltou ao início.\n");  
    break;  
default:  
    printf("Nenhuma operação.\n");  
    break;  
}
```