Lógica de Programação e Algoritmos em C

Estruturas de decisões

Quantas decisões já tomamos em nossas vidas? Algumas importantes, outras nem tanto, mas desde o momento que saímos da cama pela manhã estamos tomando decisões. Por exemplo: *"Você está em frente ao semáforo de pedestre: Se ele estiver verde, você pode atravessar a rua, caso contrário, se estiver vermelho, você deve esperar. "* E mesmo assim, quando você tomar a decisão de atravessar a rua, de uma boa olhada para ver se não vem algum motorista *"louco"*, que não vai prestar a atenção em uma sinaleira para pedestres.

Os programas de computador irão enfrentar os mesmos dilemas, não de atravessar uma rua, mas de tomar decisões dentro da sequencia de comandos existentes dentro de um código fonte.

Haverá momentos que o programa deverá tomar caminhos diferentes e estes caminhos devem ser decididos por eles mesmos.

Em um programa que leia as notas escolares de um aluno, calcule a média e apresenta o resultado, talvez devêssemos definir se o aluno foi **"Aprovado"** ou **"Reprovado"**?

Operadores relacionais

Para comparar valores e definir o caminho a ser seguido usamos os operadores relacionais.

Operador	Ação	Exemplo	Resultado
>	maior do que	6 > 2	Verdadeiro
>=	maior ou igual a	6 >= 3	Falso
<	menor do que	6 < 9	Verdadeiro
<=	menor ou igual a	6 <= 9	Verdadeiro
= =	igual a	6 = = 7	Falso
! =	diferente	6 ! = 7	Verdadeiro

Temos dois estados lógicos que são a representação da lógica "booleana", que indica: "verdadeiro" e "falso".

Na linguagem C não temos os valores booleanos: **"Verdadeiro"** ou **"Falso"**. O que representa o estado **"Falso"** é o número **"0"** e o que representa o estado **"Verdadeiro"** é o número **"1"**, ou ainda, qualquer valor diferente de zero é **"Verdadeiro"**.

ValorX	Comparação	ValorY	Resultado	Significado Booleano
5	>	8	0	falso
5	>=	8	0	falso
5	<	8	1	verdadeiro
5	<=	8	1	verdadeiro
5	==	8	0	falso
5	!=	8	1	verdadeiro

Desvio condicional simples

O comando utilizado para fazer o **"Desvio condicional simples"** é o **"if"**, que significa **"se"**.

O seu formato geral é:

```
if(expressão)
        comando1;
```

Neste **"if"** a **"expressão"** será verificada e se for **"verdadeira"** o **"comando1;"** será executado.

Quando houver mais de uma linha de código dentro do **"if"** devemos criar um bloco de comandos com o uso das chaves. Marcamos o inicio do bloco com uma chave **"{"** e finalizamos com outra chave "**}"**. No formato acima as chaves não são obrigatórias, mas no formato abaixo elas são necessárias.

O formato seria o seguinte:

```
if(expressão)
{
        comando1;
        comando2;
}
```

Neste caso a **"expressão"** é avaliada e se for **"verdadeira"** os comandos: **"comando1;"** e **"comando2;"**, serão executados.

Quando um **"if"** testa o valor de numérico de uma expressão, podemos fazer algumas abreviações:

```
if(expressão != 0)
```

Poderia ter sido escrito da seguinte forma:

```
if(expressão)
```

Para saber se uma **"expressão"** é diferente de **"0"** a segunda forma pode ser utilizada sem nenhum prejuízo.

Desvio condicional composto

Esta estrutura tem o seu formato geral da seguinte forma:

```
if(expressão)
        comando1;
else
        comando2;
```

Neste **"if"** a **"expressão"** será verificada e se for **"verdadeira"** o **"comando1;"** será executado, caso contrário, ou seja: **"se não"** for **"verdadeira"** o **"comando2"** será executado.

Se houver necessidade de mais comandos dentro da estrutura, não se esqueça de colocar estes comandos em um bloco delimitado pelas chaves: **"{"** (que começa o bloco) e **"}"** (que encerra o bloco de comandos).

O formato seria o seguinte:

```
if(expressão)
{
        comando1;
        comando2;
}
else
{
        comando3;
        comando4;
}
```

Desvio condicional encadeado

Haverá situações que teremos sucessivas verificações. Nestes casos faremos usos de vários **"if-else's"** encadeados.

O seu formato geral é:

```
if(expressão1)
        comando1;
else if(expressão2)
        comando2;
else
        comando3;
```

Nesta sequência é verificada a **"expressão1",** se ela for **"verdadeira"** será executado o **"comando1"**, senão é verificada a **"expressao2"**, se esta for **"verdadeira"** será executado o **"comando2"**, se nenhuma delas for verdadeira será executado o **"comando3"**.

Desvio condicional de múltiplos casos

Quando temos um desvio condicional encadeado, em alguns casos, podemos fazer uso da estrutura chamada **"switch"**, que representa a estrutura de desvio condicional de múltiplos casos. Esta estrutura tem um desenho mais elegante que a estrutura encadeada. Ela pode ser utilizada sempre que você estiver fazendo uma sequência de comparações de igualdade.

O seu formato geral é:

```
switch (variavel)
{
case :
        break;
case :
        break;
case :
        break;
default:
        break;
}
```