

Aula 10

Comandos de Repetição

Algoritmos e Programação de Computadores
Profs: Ronaldo Castro de Oliveira – ronaldo.co@ufu.br
Anilton Joaquim da Silva – anilton@ufu.br

Introdução

- Em certas situações é necessária a repetição de um conjunto de comandos. Em situações como esta, temos duas opções: ou copiamos e colamos todo o trecho que desejamos repetir, fazendo os ajustes necessários; ou utilizamos uma saída mais inteligente por meio de comandos especiais que permitem automatizar a repetição.

O comando **while** - enquanto

- Forma geral:

```
while (<condição>)
```

```
{
```

```
    //bloco de comandos a ser repetido
```

```
}
```

- O bloco de comandos entre as chaves será repetido enquanto a condição dentro dos parênteses for verdadeira.

OBS: Se a condição lógica falhar inicialmente o bloco de comando não é executado nenhuma vez.

Exemplo - while

Programa que lê duas notas e imprime a média da nota de N alunos.

```
#include <iostream>

using namespace std;

float calcula_media(float nota1, float nota2)
{
    return (nota1 + nota2)/2.0;
}

int main()
{
    int numero, contador;
    float nota1, nota2, media;
    cout << "Programa de media de notas de N alunos." << endl;
    cout << "Entre com o numero de alunos: ";
    cin >> numero;

    contador = 1;
    while (contador <= numero)
    {
        cout << endl << "Entre com 1 nota do aluno " << contador <<": ";
        cin >> nota1;
        cout << "Entre com 2 nota do aluno " << contador <<": ";
        cin >> nota2;
        media = calcula_media(nota1, nota2);
        cout << "Media do aluno " << contador <<" e igual a: " << media << endl;
        contador += 1;
    }
    return 0;
}
```

Exemplo - **while**

- Soma os números pares entre 1 e 999.

...

```
int n = 2, // primeiro par maior do que 1
```

```
soma = 0; // soma inicialmente zerada
```

```
while (n < 999)
```

```
{
```

```
    soma = soma + n;
```

```
    n = n + 2;
```

```
}
```

```
cout << "O valor da soma eh " << soma << endl;
```

...

O comando **do - while**

- Forma geral:

```
do  
{
```

```
    //bloco de comandos a ser repetido
```

```
} while (<condição>);
```

- O bloco de comandos entre as chaves será repetido enquanto a condição dentro dos parênteses for verdadeira.

OBS: o bloco de comando é executado pelo menos uma vez.

Exemplo do-while

Programa que lê duas notas e imprime a média da nota de N alunos.

```
#include <iostream>
using namespace std;

float calcula_media(float nota1, float nota2)
{
    return (nota1 + nota2)/2.0;
}

int main()
{
    int numero, contador;
    float nota1, nota2, media;
    cout << "Programa de media de notas de N alunos." << endl;
    cout << "Entre com o numero de alunos: ";
    cin >> numero;

    contador = 1;
    do
    {
        cout << endl << "Entre com 1 nota do aluno " << contador <<": ";
        cin >> nota1;
        cout << "Entre com 2 nota do aluno " << contador <<": ";
        cin >> nota2;
        media = calcula_media(nota1, nota2);
        cout << "Media do aluno " << contador <<" e igual a: " << media << endl;
        contador += 1;
    } while (contador <= numero);
    return 0;
}
```

Exemplo do-while

- Programa que lê 10 números e imprime o maior:

```
...
int i = 0; // contador da qtde de numeros lidos
int maior, n;
do
{
    cout << "Entre um numero: ";
    cin >> n;
    if (i == 0) // se for o primeiro numero lido
    { //ele sera o maior
        maior = n;
    }
    else // a partir do segundo
    {
        if(n > maior) // atualizo o maior
        {
            maior = n;
        }
    }
    i = i + 1;
} while (i < 10);
```

...

Exemplo do-while

- Lê um determinado valor dentro de uma faixa de valores esperados, por exemplo uma data:

```
...
int dia, mes;
do
{
    cout << "Digite o dia (entre 1 e 31): "
    cin >> dia;
} while ((dia < 1) || (dia >31));
do
{
    cout << "Digite o mes (entre 1 e 12): "
    cin >> mes;
} while ((mes <1) || (mes >12));
...
```

O comando **for**

- Regra geral

```
for (DI; C; I)
```

```
{
```

```
//Bloco de comandos a serem repetidos
```

```
}
```

- O comando **for** tem três partes em sua declaração, além dos comandos a serem repetidos.
 - **DI** – as variáveis podem ser **D**eclaradas e **I**niciadas. Variáveis já existentes também podem ter seus valores ajustados em DI;
 - **C** – define a **Condição** necessária à execução do bloco de comandos. Enquanto a condição for verdadeira, o bloco será executado.
 - **I** – comandos de modificação de variáveis, como Incremento e decremento, são colocados diretamente na declaração do for. O comando é executado ao final de cada iteração.

O comando **for**

- A execução do **for** segue os seguintes passos:
 1. Iniciação (execução de **DI**) – inicializa a variável controladora da repetição;
 2. Avaliação (teste da condição em **C**) – avalia o resultado lógico de comparação da variável controladora;
 3. Execução do bloco de comandos – é executado se a condição **for** verdadeira;
 4. Incremento – somente após a execução é que o função executa o incremento da variável controladora;
 5. Volta ao passo 2

Exemplo de **for**

- Soma os números pares entre 1 e 999

...

```
int soma = 0; // soma inicialmente zerada
```

```
for (int n = 2; n < 999; n += 2)
```

```
{
```

```
    soma = soma + n;
```

```
}
```

```
cout << "O valor da soma e: " << soma <<  
endl;
```

...

Exemplo de **for**

- **Programa que lê 10 números e imprime o maior:**

```
...
int i;           // contador da qtde de numeros lidos
int maior, n;
for(i = 0; i < 10; i++)
{
    cout << "Entre com um numero: ";
    cin >> n;
    if ((i == 0) || (n > maior)) // inicializo e atualizo o maior
    {
        maior = n;
    }
}
...
```

Exemplo de **for** – declarações especiais

- Função menu de opções:

```
int menu()
{
    int opcao = 0;
    for( ; opcao < 1 || opcao > 4 ; )
    {
        cout << "1 - Soma" << endl << "2 - Media" << endl
            << "3 - Menor" << endl << "4 - Maior" << endl;
        cout << "Qual sua opcao? ";
        cin >> opcao;
        if (opcao < 1 || opcao > 4)
            cout << "Opcao invalida!!!!" << endl;
    }
    return opcao;
}
```

Exemplo de **for** – declarações especiais

- Função menu de opções – melhorando um pouco mais – usando **break**:

```
int menu()
{
    int opcao = 0;
    for( ; ; )
    {
        cout << "1 - Soma" << endl << "2 - Media" << endl
             << "3 - Menor" << endl << "4 - Maior" << endl;
        cout << "Qual sua opcao? ";
        cin >> opcao;
        if (opcao > 0 && opcao < 5)
            break;
        else
            cout << "Opcao invalida!!!!" << endl;
    }
    return opcao;
}
```