The background of the slide is an abstract, high-contrast image of numerous 3D cubes or rectangular blocks. These blocks are arranged in a way that creates a sense of depth and perspective, with some blocks appearing to rise from a dark, textured surface. The lighting is dramatic, casting deep shadows and highlighting the edges of the blocks, which are in various shades of gray and black.

Profª Cláudia
Werlich

ESTRUTURA DE REPETIÇÃO FOR

Estrutura de Repetição

repetição

1. ato ou efeito de repetir
2. ato ou efeito de voltar a fazer ou dizer algo já feito ou dito; iteração

Estrutura de Repetição



Estrutura de Repetição



Estrutura de Repetição

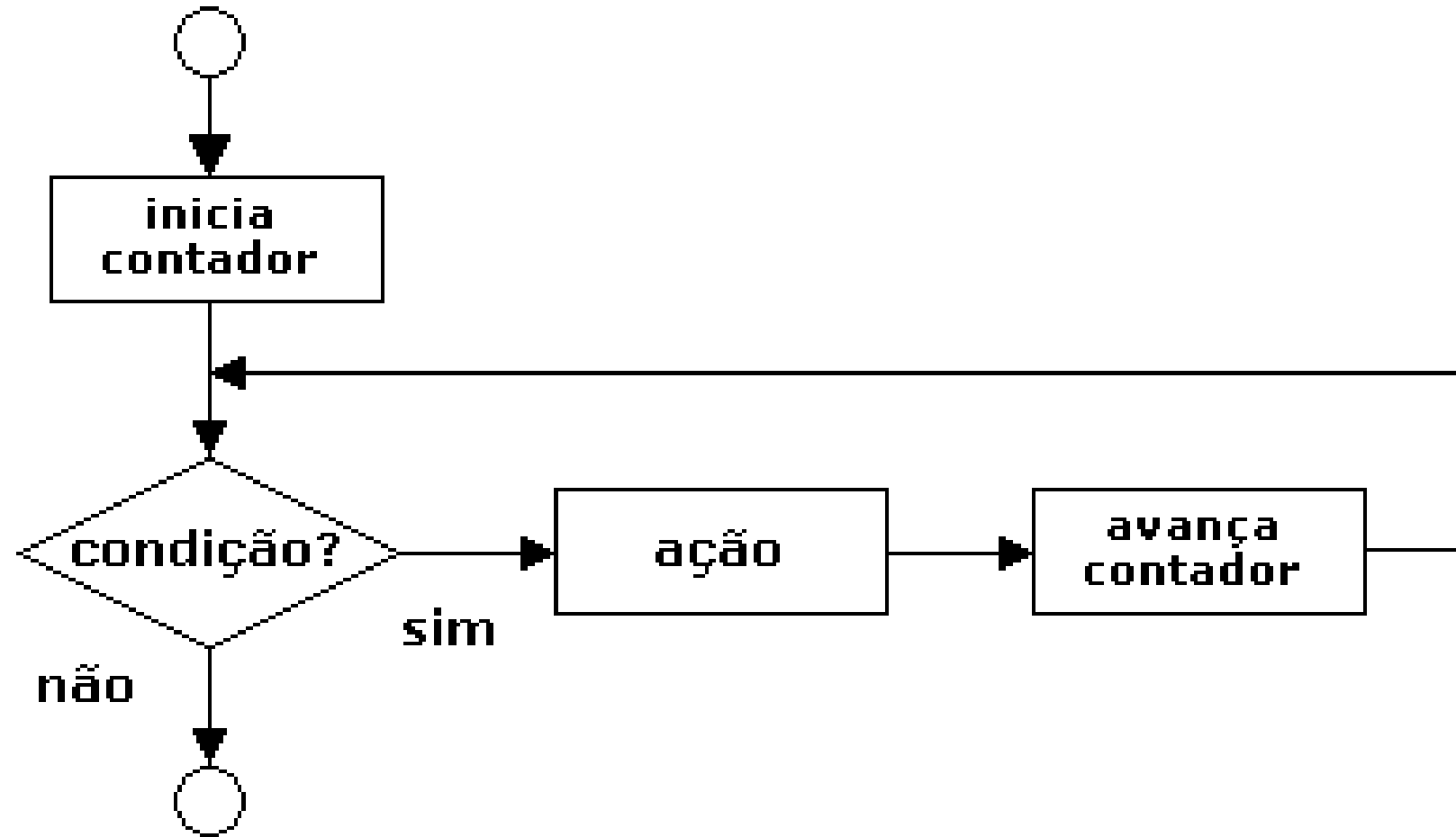


Estrutura de Repetição

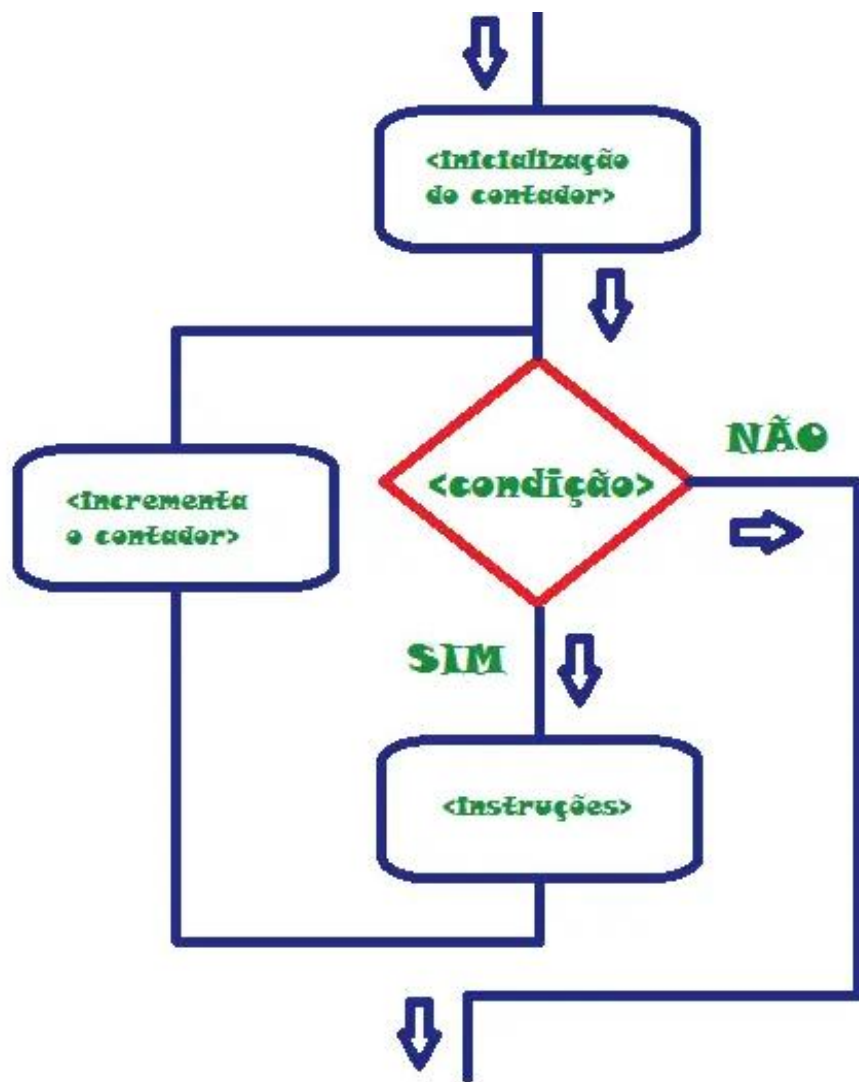
As estruturas de repetição permitem executar mais de uma vez um mesmo trecho de código.

É uma forma de executar blocos de comandos somente sob determinadas condições, mas com a opção de repetir o mesmo bloco quantas vezes for necessário.

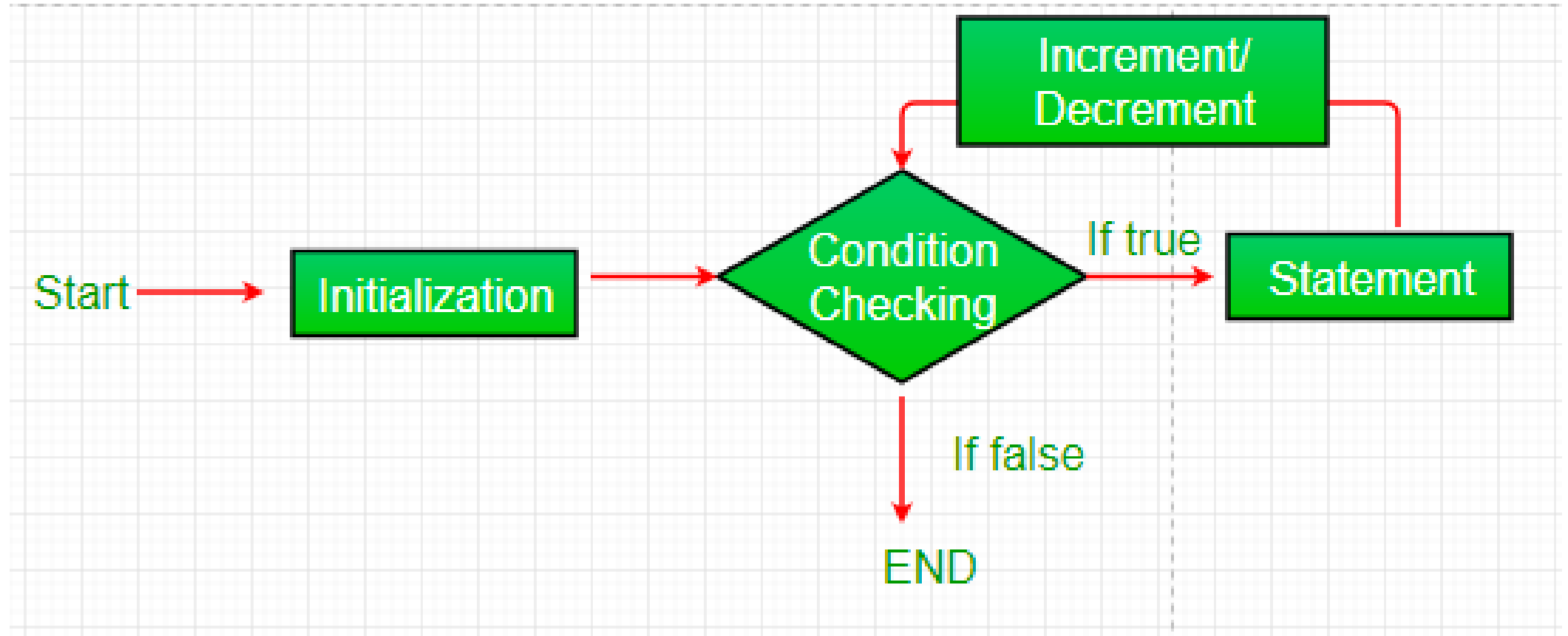
Para - faça == for



Para - faça == for



Para - faça == for



```
for (int i=0; i<10; i++)
```

Initialization

Condition

Iteration

COMANDO for NO C++

COMANDO for NO C++

parenthesis

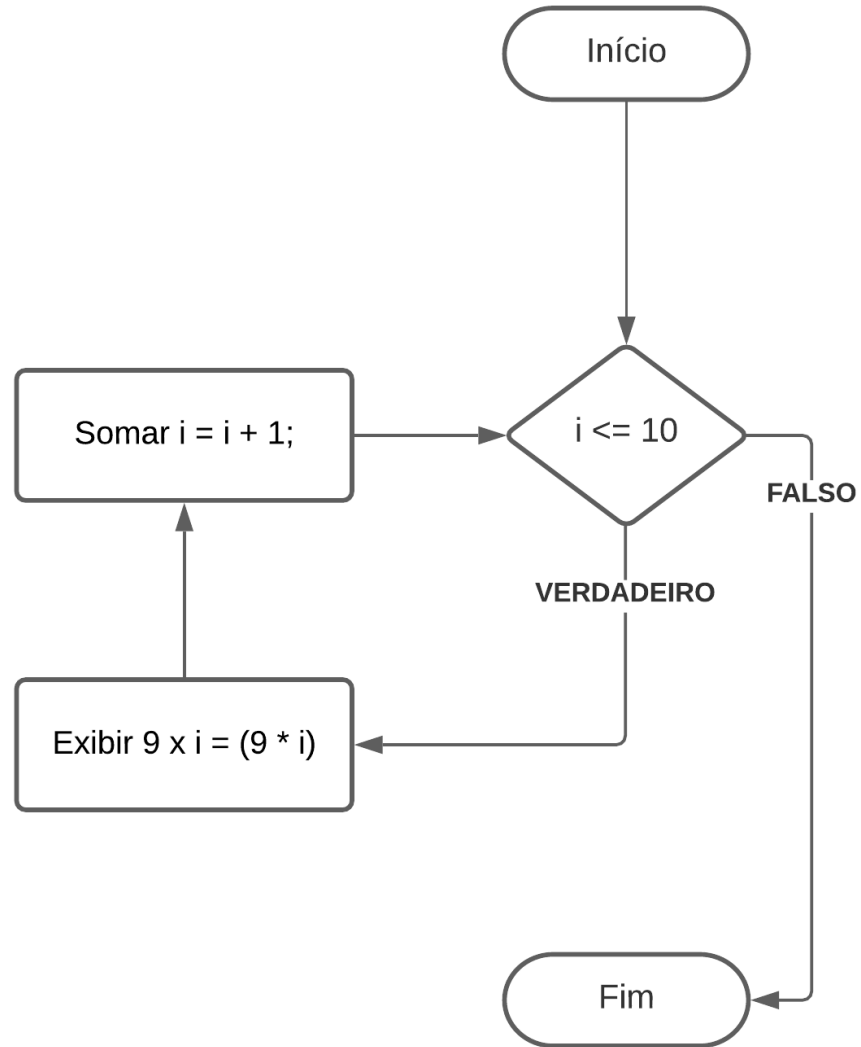
declare variable (optional)

initialize test increment or decrement

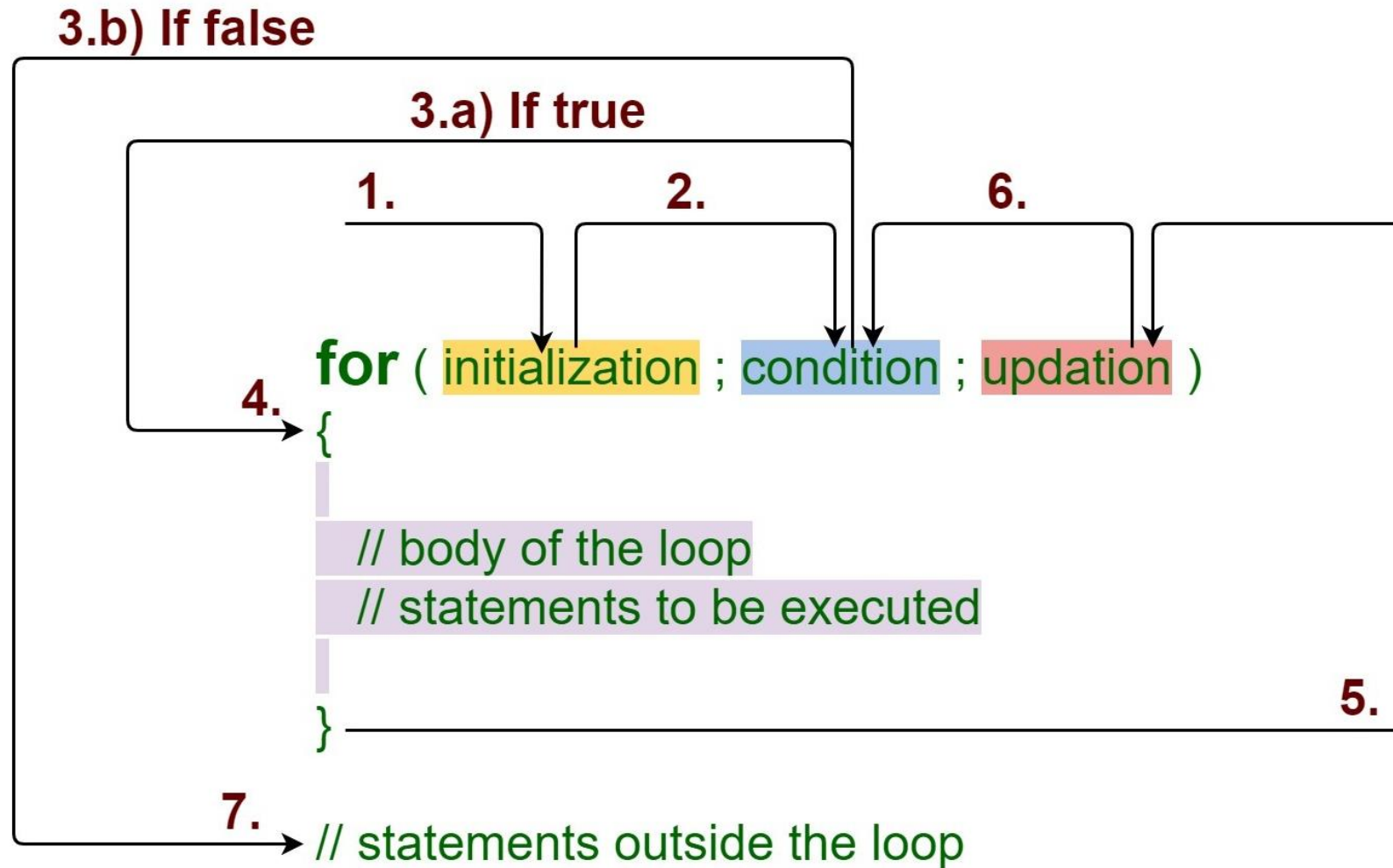
```
for(int x = 0; x < 100; x++) {  
    println(x);    // prints 0 to 99  
}
```

The diagram illustrates the structure of a C++ for loop. It features a code snippet: `for(int x = 0; x < 100; x++) { println(x); // prints 0 to 99 }`. Above the code, four labels are positioned: 'parenthesis' (above the opening brace), 'declare variable (optional)' (above 'int x'), 'initialize' (above '= 0'), 'test' (above 'x < 100'), and 'increment or decrement' (above 'x++'). Colored arrows point from these labels to their respective parts in the code: a blue arrow from 'parenthesis' to the opening brace, a red arrow from 'declare variable (optional)' to 'int x', a green arrow from 'initialize' to '= 0', a green arrow from 'test' to 'x < 100', and a green arrow from 'increment or decrement' to 'x++'. A long blue curved arrow also originates from the 'parenthesis' label and points to the closing brace of the loop.

COMANDO for NO C++



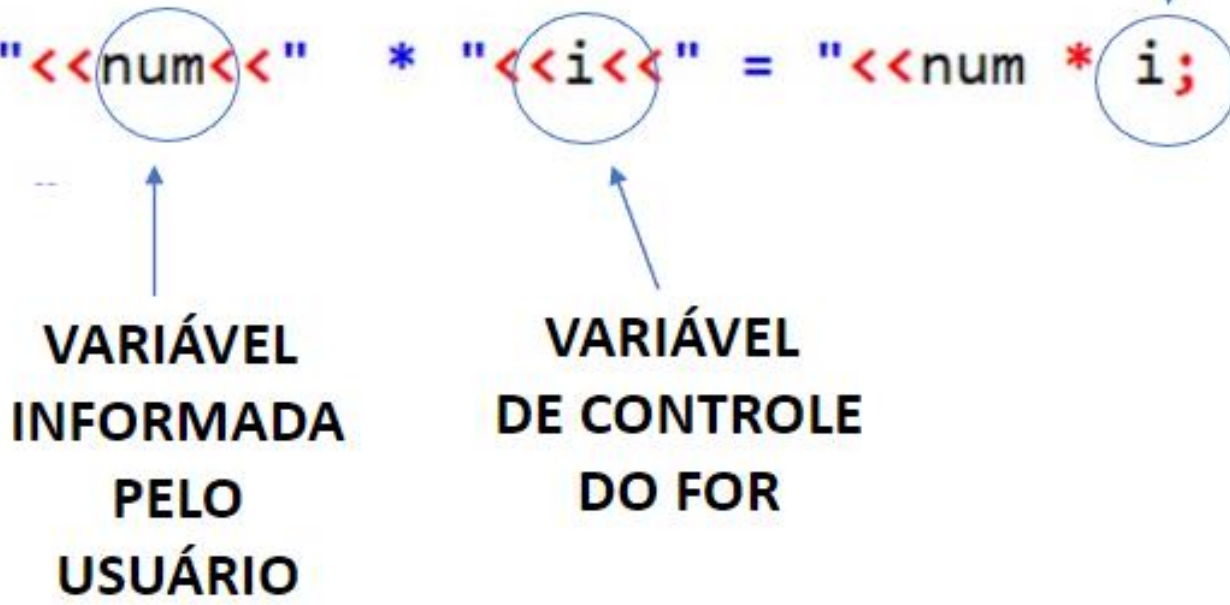
COMANDO for NO C++



COMANDO for NO C++

VARIÁVEL
DE CONTROLE
DO FOR

```
16 cout<<" \n\n Informe um número: ";
17 cin>>num;
18 for(i=0;i<=10;i++)
19 {
20     cout<<"\n " << num << " * " << i << " = " << num * i;
21 }
```



VARIÁVEL
INFORMADA
PELO
USUÁRIO

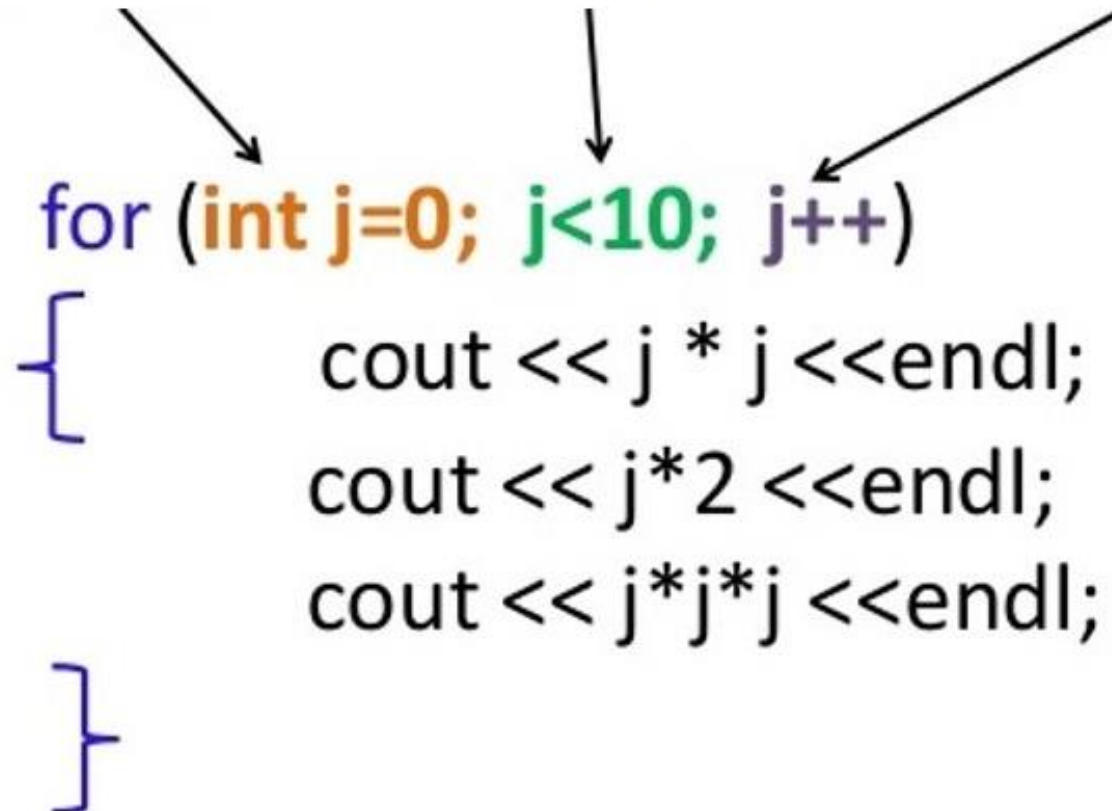
VARIÁVEL
DE CONTROLE
DO FOR

COMANDO for NO C++ - INCREMENTO

VARIÁVEL DE
INICIALIZAÇÃO

CONDIÇÃO

INCREMENTO DA VARIÁVEL



```
for (int j=0; j<10; j++)  
{  
    cout << j * j << endl;  
    cout << j*2 << endl;  
    cout << j*j*j << endl;  
}
```

The diagram illustrates the three components of a C++ for loop: 'VARIÁVEL DE INICIALIZAÇÃO' (Initialization Variable) points to 'int j=0', 'CONDIÇÃO' (Condition) points to 'j<10', and 'INCREMENTO DA VARIÁVEL' (Variable Increment) points to 'j++'. The code block shows the loop structure with three cout statements inside the body.

COMANDO for NO C++ - DECREMENTO

INICIALIZAÇÃO

CONDIÇÃO

```
cout<<"\n Insira um valor para calcular o fatorial: ";  
cin>>num;  
for(fat = 1; num > 1; num--)  
{  
    fat = fat * num;  
}
```

DECREMENTO

COMANDO for NO C++

- ✓ A Variável de controle do for **DEVE** ser do tipo **INT**.
- ✓ A variável **PRECISA** ser **INICIALIZADA** no início.
- ✓ A variável é inicializada **SOMENTE UMA VEZ** na estrutura.
- ✓ Cuidados com os **PONTO E VÍRGULAS** no comando for.
- ✓ Não esqueça de **USAR SEMPRE** as **CHAVES**.

Tabuada usando o comando for

```
5  #include <iostream> // BIBLIOTECA PADRAO C++
6  #include <iomanip> // PARA O setlocale
7  using namespace std; // OBRIGATÓRIO
8  int i,num;
9  main()
10 {
11     setlocale(LC_ALL, "Portuguese");
12     // Entrada dos Dados
13     cout<<" \n===== ";
14     cout<<" \n\t PROGRAMA TABUADA COM O FOR ";
15     cout<<" \n===== ";
16     cout<<" \n\n Informe um número: ";
17     cin>>num;
18     for(i=0;i<=10;i++)
19     {
20         cout<<"\n " << num <<" * " << i <<" = " << num * i;
21     }
22     cout<<"\n\n\n";
23 }
```

Exemplo do uso do comando `for`

Faça um programa que imprima na tela todos os números de 0 a 100.

Observe que:

- ❖ Não há solicitação de valores de entradas.
- ❖ O programa deverá de forma direta imprimir todos os números de 0 a 100.

Exemplo do uso do comando **for**

```
6  #include <iostream>
7  #include <iomanip>
8  using namespace std;
9  int i;
10 main()
11 {
12     setlocale(LC_ALL, "Portuguese");
13     cout<<" \n===== ";
14     cout<<" \n\t PROGRAMA NÚMEROS DE 0 a 100 ";
15     cout<<" \n===== \n";
16     for(i=0;i<=100;i++)
17     {
18         cout<<i<<" - ";
19     }
20     cout<<"\n\n\n";
21 }
```


Exemplo do uso do comando for

**COMO?
MUDAR
PARA**



```
6  #include <iostream>
7  #include <iomanip>
8  using namespace std;
9  int i;
10 main()
11 {
12     setlocale(LC_ALL, "Portuguese");
13     cout<<" \n===== ";
14     cout<<" \n\t PROGRAMA NÚMEROS DE 0 a 100 ";
15     cout<<" \n===== \n";
16     for(i=0;i<=100;i++)
17     {
18         cout<<i<<" - ";
19     }
20     cout<<"\n\n\n";
21 }
```

Exemplo do uso do comando for

```
6  #include <iostream>
7  #include <iomanip>
8  using namespace std;
9  int i;
10 main()
11 {
12     setlocale(LC_ALL, "Portuguese");
13     cout<<" \n===== ";
14     cout<<" \n\t PROGRAMA NÚMEROS DE 0 a 1000 ";
15     cout<<" \n===== \n";
16     for(i=0;i<=1000;i++)
17     {
18         cout<<i<<" - ";
19     }
20     cout<<"\n\n\n";
21 }
```

1000

Exemplo da Lista de Números Pares

```
6  #include <iostream>
7  #include <iomanip>
8  using namespace std;
9  int i;
10 main()
11 {
12     setlocale(LC_ALL, "Portuguese");
13     cout<<" \n===== ";
14     cout<<" \n\t PROGRAMA PARES DE 0 a 100 ";
15     cout<<" \n===== \n";
16     for(i=0;i<=100;i++)
17     {
18         if (i%2==0)
19             cout<<i<<" - ";
20     }
21     cout<<"\n\n\n";
22 }
```



EXERCÍCIOS

