

Aula - Estruturas Condicionais C++

BIBLIOTECA

USAREMOS `#include<bits/stdc++.h>`

✓ **COM** `using namespace std;`

```
#include<bits/stdc++.h>

using namespace std;

int main() {
    cout << "Olá, mundo!" << endl;
    return 0;
}
```

Explicação:

- O `using namespace std;` diz ao compilador que **vamos usar nomes do espaço de nomes (namespace) padrão (`std`) sem precisar escrevê-lo explicitamente.**
- Isso permite escrever `cout` ao invés de `std::cout`, e `endl` ao invés de `std::endl`.

✗ **SEM** `using namespace std;`

```
#include<bits/stdc++.h>

int main() {
    std::cout << "Olá, mundo!" << std::endl;
    return 0;
}
```

Explicação:

- Aqui, **não usamos** `using namespace std;`, então **é necessário prefixar tudo que vem do namespace** `std` **com** `std::`.
- Isso torna o código mais explícito, o que pode ser preferível em projetos maiores para evitar conflitos de nomes.



Para rodar C++ no terminal, é preciso compilar primeiro usando:
`g++ [arquivo].cpp`

Com isso, haverá a criação de um executável `a.out` - que poderá ser executado:

`./a.out`

Estruturas Condicionais

E vamos para as boas e velhas condições em programação, só que agora em C++:

IF

```
#include <bits/stdc++.h>
using namespace std;

int main(){
    int a = 50, b = 25;
    if (a < b) {
        cout << "a é menor do que b" << endl;
    } else if (a == b) {
        cout << "a é igual a b" << endl;
    }
    else {
        cout << "a é maior do que b" << endl;
    }
}
```

```
    return 0;
}
```

SWITCH

```
#include<bits/stdc++.h>
using namespace std;

int main(){
    int opcao;
    cout << "digite um numero para opção: 1 para opc1 ou 2 para opc2" << endl;
    cin >> opcao;

    switch(opcao) {
        case 1:
            cout << "Ebaaa. Você escolheu a opção 1" << endl;
            break;
        case 2:
            cout << "Okaaay. Você escolheu a opção 2" << endl;
            break;
        default:
            cout << "Digite 1 ou 2 para selecionar as respectivas opções, por favor." << endl;
            break;
    }
    return 0;
}
```

Operador condicional ?: (também conhecido como Operador ternário)

```
#include<bits/stdc++.h>

using namespace std;

int main() {
    int j = 2, k = 3;
```

```
cout << (j > k ? j : k) << " é maior" << endl;  
return 0;  
}
```