

# ***Laboratório***

## ***Algoritmos e***

## ***Estruturas de dados I***

Estrutura Condicional – Parte 2

Ramon – 7º Período – Engenharia da Computação

### ***Roteiro***

- ***Operação Ternária***
- ***Estrutura SWITCH***
- ***Exercícios***

## **Operação Ternária**

Uma operação ternária é uma operação com três operandos.

A linguagem C/C++ possui um operador chamado ternário **?**, que representa uma expressão condicional.

<condição> **?** <operação 1> : <operação 2>

**Condição:** Expressão condicional.

**Operação 1:** Comando caso a condição for verdadeira.

**Operação 2:** Comando caso a condição for falsa.

## **Exemplo**

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

int main()
{
    int ano_atual, ano_nasc, idade;

    cout << "Entre com o ano de nascimento: ";
    cin >> ano_nasc;
    cout << "Entre com o ano atual: ";
    cin >> ano_atual;

    idade = ano_atual - ano_nasc;

    cout << ((idade>=18) ? "Voce eh maior de idade!" : "Voce eh menor de idade!");

    cin.ignore();
    cin.get();
}
```

## Exemplo

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

int main()
{
    char nome1[20], nome2[20];
    int ano_nasc1, ano_nasc2, idade1, idade2;

    //cin.ignore();
    cout << "Entre com os nomes: ";
    cin.getline(nome1, 20);
    cin.getline(nome2, 20);
    cout << "Entre com os anos de nascimento: ";
    cin >> ano_nasc1 >> ano_nasc2;

    idade1 = 2011 - ano_nasc1;
    idade2 = 2011 - ano_nasc2;

    cout << (((ano_nasc1>=1980) && (ano_nasc1<=2010)) ? nome1 : "\n");
    cout << (((ano_nasc2>=1980) && (ano_nasc2<=2010)) ? nome2 : "\n");
    ((idade1 >= idade2) ? cout << "\nOrdem decrescente\n" << nome1 << nome2 : cout << "\nOrdem decrescente\n" << nome2 << nome1);

    cin.ignore();
    cin.get();
}
```

## Estrutura SWITCH

É uma estrutura condicional seletiva que permite a execução de um conjunto de instruções a partir pontos diferentes conforme o resultado de uma expressão inteira de controle.

```
switch (variável)
{
    case A: comandos;
            break;
    case B: comandos;
            break;
    case C: comandos;
            break;

    default: comandos; //opcional
            break;
}
```

## Estrutura SWITCH

### Observações:

1. Os cases não aceitam operadores lógicos. Portanto, não é possível fazer uma comparação. Isso limita o case a apenas valores definidos.
2. O switch executa seu bloco em cascata. Ou seja, se a variável indicar para o primeiro case e dentro do switch tiver 5 cases, o switch executará todos os outros 4 cases a não ser que utilizemos o comando para sair do switch. (Nos referimos ao **BREAK**).

## Exemplo

```
switch (codigo)
{
    case 1: cout << "\nNome:      " << nome << endl;
            cout << "Cargo:      Escrituario" << endl;
            cout << "Percentual: 50%" << endl;
            cout << "Salario novo: R$ " << (salario*1.5);
            break;

    case 2: cout << "\nNome:      " << nome << endl;
            cout << "Cargo:      Secretario" << endl;
            cout << "Percentual: 35%" << endl;
            cout << "Salario novo: R$ " << (salario*1.35);
            break;

    case 3: cout << "\nNome:      " << nome << endl;
            cout << "Cargo:      Caixa" << endl;
            cout << "Percentual: 22%" << endl;
            cout << "Salario novo: R$ " << (salario*1.22);
            break;

    case 4: cout << "\nNome:      " << nome << endl;
            cout << "Cargo:      Gerente" << endl;
            cout << "Percentual: 10%" << endl;
            cout << "Salario novo: R$ " << (salario*1.1);
            break;

    case 5: cout << "\nNome:      " << nome << endl;
            cout << "Cargo:      Diretor" << endl;
            cout << "Percentual: Nao ha aumento!" << endl;
            cout << "Salario novo: R$ " << salario;
            break;
}
```