

Estruturas condicionais- senao se

O **senao se** é utilizado quando precisamos adicionar mais de uma condição a estrutura **se/senao**, veja o exemplo.

```
inteiro idade = 19

//O comando se verifica se a idade informada é maior ou igual a 18
se (idade > 18)
{
    //Se for, escreve "Tem mais de 18 anos"
    escreva("Tem mais de 18 anos")
}
//Se não for, verifica se a idade é menor que 18
senao se(idade < 18)
{
    //Se for, escreve "Tem menos de 18 anos"
    escreva("Tem menos de 18 anos")
}
//Se nenhuma das verificações for atendida executa o senao
senao
{
    //escreve "Tem exatamente 18 anos"
    escreva("Tem exatamente 18 anos")
}
```

Se executarmos esse código em Portugol ele imprimirá duas mensagens na tela: “Tem mais de 18 anos” e “Tem exatamente 18 anos”

```
inteiro idade = 19

se (idade > 18)
{
    escreva("Tem mais de 18 anos")
}
se(idade < 18)
{
    escreva("Tem menos de 18 anos")
}
senao
{
    escreva("Tem exatamente 18 anos")
}
```

Isso ocorre, pois quando utilizamos múltiplos **se** eles são executados em blocos de forma independente. Os blocos são executados em ordem, um de cada vez.

```

inteiro idade = 19

se (idade > 18)           Bloco 1
{
    escreva("Tem mais de 18 anos")
}

se(idade < 18)           Bloco 2
{
    escreva("Tem menos de 18 anos")
}
senao
{
    escreva("Tem exatamente 18 anos")
}

```

Desse modo, como a idade é 19, a condição retornará como verdadeiro no primeiro bloco e falso no segundo. Para resolver isso utilizamos o `senao se`, pois ele vai fazer com que toda estrutura se torne um único bloco. Veja:

```

inteiro idade = 19

se (idade > 18)           Bloco único
{
    escreva("Tem mais de 18 anos")
}
senao se(idade < 18)
{
    escreva("Tem menos de 18 anos")
}
senao
{
    escreva("Tem exatamente 18 anos")
}

```

Dessa forma a estrutura será executada como se um único bloco e o `senao` só será executado se todas as condições anteriores falharem.