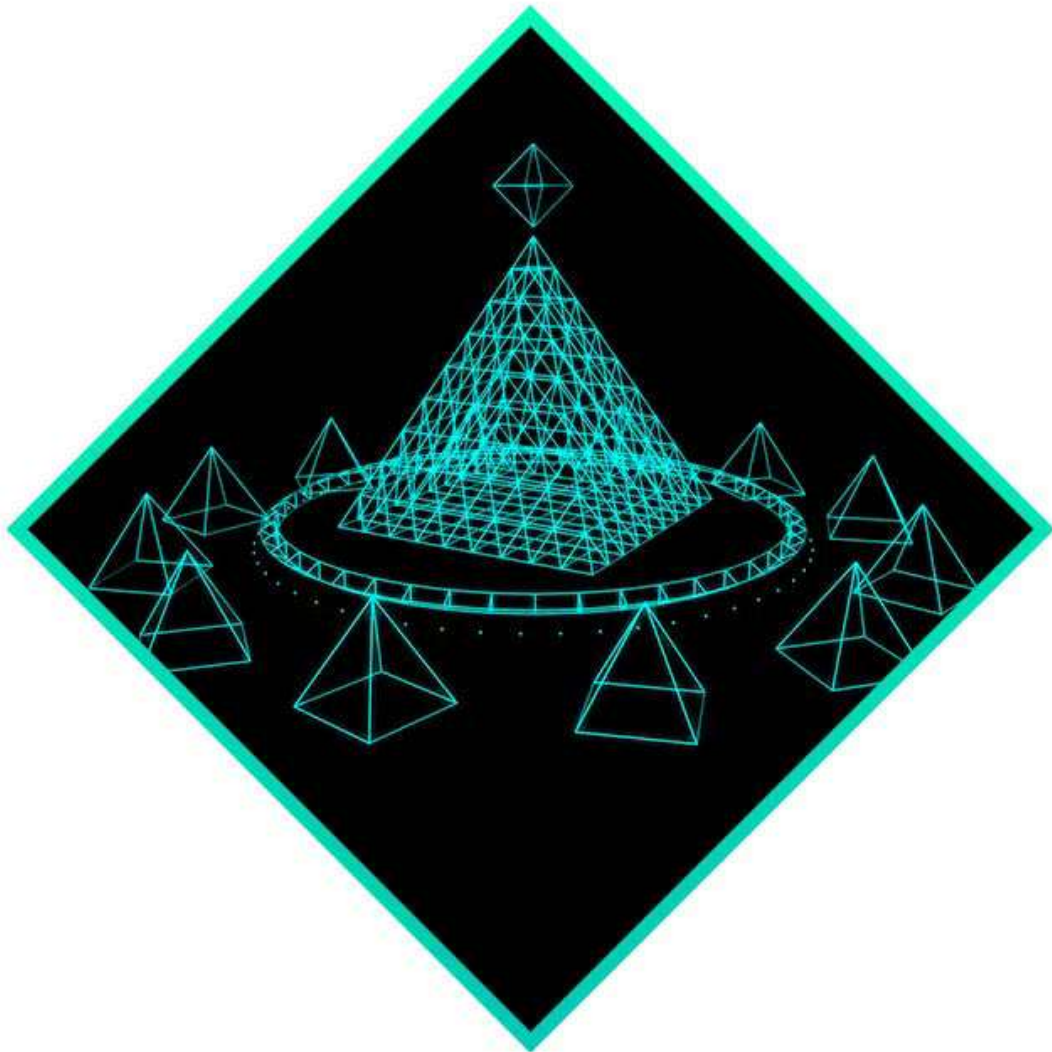


COMANDOS CONDICIONAIS SIMPLES

Estrutura de decisão

Conceito

No desenvolvimento de aplicações, estruturamos nossos algoritmos para realizar determinadas atividades. Uma destas estruturas é chamada **estrutura de decisão**. Esta estrutura permite que a aplicação possa ter mais de uma sequência a ser seguida, que será decidida a partir da inferência de um determinado valor. Este valor, na Linguagem C, será do tipo booleano, portanto, os valores poderão ser falsos (quando forem zero ou *null*) ou verdadeiros (nos demais casos).



Esta estrutura é conhecida como **SE-ENTÃO** e **SE-ENTÃO-SENÃO**. No primeiro caso, um determinado segmento de código será executado somente se a expressão lógica, que será inferida, for verdadeira, enquanto no segundo

caso a inferência da expressão lógica determinará qual segmento de código será executado.

Graficamente, esta estrutura é representada por um losango, em que cada quina representa:



A decisão corresponde a uma expressão ou variável cujo valor será analisado, conforme já citado.

Exemplo

Expressões que são verdadeiras. Considere em todos os exemplos que a variável **a** é inteira e tem valor 1:

- a. SE (a)
- b. SE (a == 1)
- c. SE (a > 0)

Na Linguagem C, o SE é representado pela palavra reservada *if*, assim os itens acima seriam representados nesta linguagem como:

- a. if(a)
- b. if(a==1)
- c. if(a > 0)

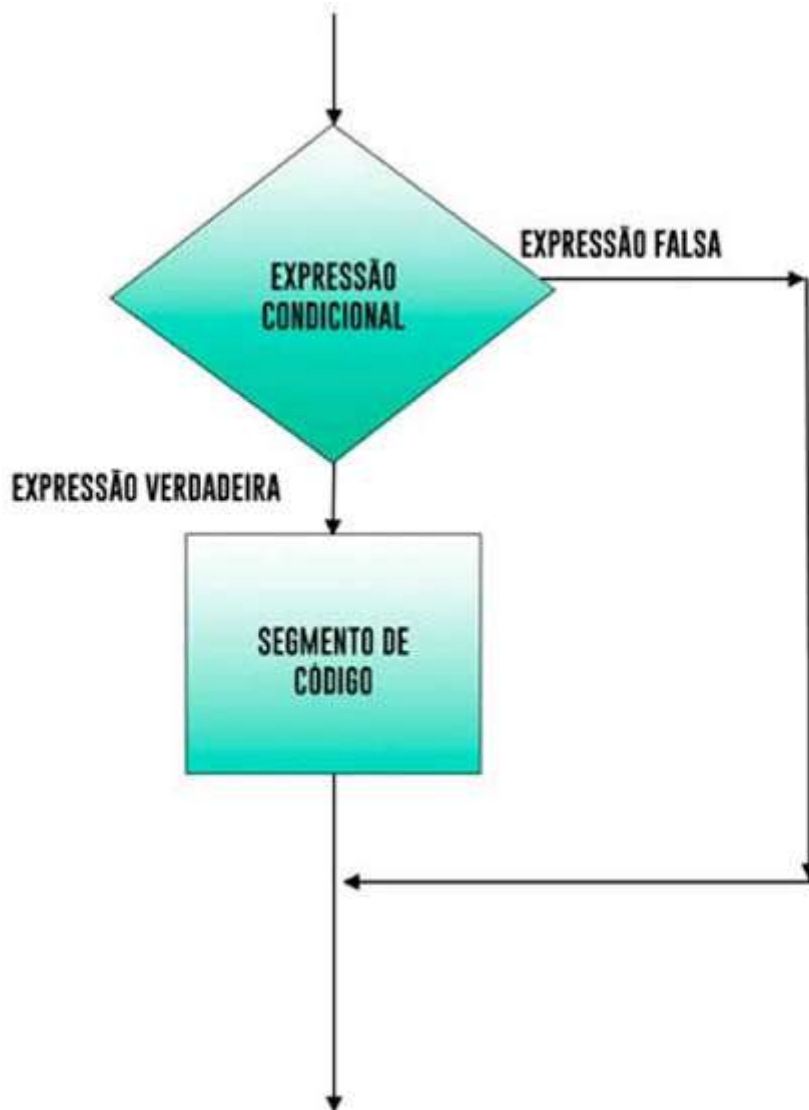
É importante relembrar que a **Linguagem C é sensível ao contexto**, assim deve-se usar *if* com todos os caracteres minúsculos. Comandos como *If*, *iF* e *IF* não são válidos. Além disso, os parênteses utilizados acima são obrigatórios.

Mais adiante, falaremos sobre quando tivermos uma estrutura destas dentro de outra estrutura, assim chamada de aninhada, e quando duas estruturas estão posicionadas de forma sequencial, denominada de encadeada.

Estrutura de decisão simples

Estrutura de decisão simples é caracterizada pela verificação de uma expressão lógica e, caso ela atenda aos requisitos estipulados, que neste caso será a expressão ser verdadeira, um determinado segmento de código é executado. Caso seja falso, ou seja, valor igual a zero, nulo ou vazio, nenhum segmento de código é executado e o programa continua no comando que sucede ao *if*.

Graficamente, esta estrutura pode ser apresentada através da figura:



No losango, que representa a estrutura de decisão, temos a **expressão lógica**. Esta expressão será avaliada. Caso o seu valor seja verdadeiro, um bloco de comandos (segmento de código) é executado logo após a expressão ter sido calculada. Caso contrário, será dada continuidade à execução da aplicação.

Na tabela a seguir, do lado esquerdo é exibido este código representado em PORTUGOL, e do lado direito o mesmo código, usando a Linguagem C.

PORTUGOL	LINGUAGEM C
SE EXPRESSÃO_CONDICIONAL	if(expressao_condicional)
ENTÃO	{
BLOCO DE COMANDOS	bloco_de_comandos;
FIM_ENTÃO	}
FIM_SE	



Na primeira linha é apresentado o início do comando. No Portugol, a expressão SE representa o comando *if*. A expressão condicional é apresentada obrigatoriamente entre parênteses. É recomendável que não haja espaço entre a palavra reservada *if* e o parêntese. Todavia é permitido, e até aconselhável, que haja espaço entre a expressão condicional e os parênteses.

PORTUGOL	LINGUAGEM C
SE EXPRESSÃO_CONDICIONAL	if(expressao_condicional)
ENTÃO	{
BLOCO DE COMANDOS	bloco_de_comandos;
FIM_ENTÃO	}
FIM_SE	



Na linha seguinte é apresentada a palavra ENTÃO, que interpretamos como o início do bloco de comandos, que na Linguagem C é representado pelo sinal de abre chaves {.

PORTUGOL	LINGUAGEM C
SE EXPRESSÃO_CONDICIONAL	if(expressao_condicional)
ENTÃO	{
BLOCO DE COMANDOS	bloco_de_comandos;
FIM_ENTÃO	}
FIM_SE	



O bloco de comandos é apresentado na sequência, porém, na Linguagem C, caso haja apenas um comando neste bloco, o uso das chaves torna-se opcional. No caso de existirem duas ou mais instruções, estas chaves se tornam obrigatórias, uma vez que poderia haver confusão com o caso anterior.

Exemplo 1

Normalmente, ao preencheremos formulários na web, somos questionados se desejamos receber mais informação a respeito daquele assunto. Como isto poderia ser representado na parte da implementação do código?

```
int Flag_Deseja_Receber_Mais_Informacoes;
if (Flag_Deseja_Receber_Mais_Informacoes){
    Enviar_Mais_Informacoes();
}
```

Nossa expressão condicional é uma variável do tipo inteira: Flag_Deseja_Receber_Mais_Informacoes. Assim, ela recebe valores inteiros.

Já vimos que na Linguagem C é considerado falso se a variável possui valores 0, *null* ou vazio; e é considerado verdadeiro no caso contrário, ou seja, se é diferente dos valores 0, *null* e vazio, dado o tipo de dado.

Neste caso, a variável é inteira, então, caso o valor dela seja 0, será considerado falso, caso possua qualquer outro valor (positivo ou negativo), será considerado verdadeiro. Outra expressão condicional equivalente poderia ser colocada neste *if*, porém, o mais usual é como foi apresentado. Um exemplo equivalente seria Flag_Deseja_Receber_Mais_Informacoes!=0.

Exemplo 2

Veremos agora um exemplo simples da estrutura de decisão condicional simples, para tal o programa em C abaixo pede a média de um aluno hipotético e decide se o aluno foi aprovado ou não, imprimindo uma mensagem na tela.

```
#include <stdio.h>

int main (void)
{
    float nota;

    printf ("Digite a nota: ");
    scanf("%f", &nota);
    printf("%f\n",nota);

    if (nota >= 5)
        printf("Parabéns você foi aprovado\n");
    if (nota < 5)
        printf("Não deu desta vez, mas não desista!");

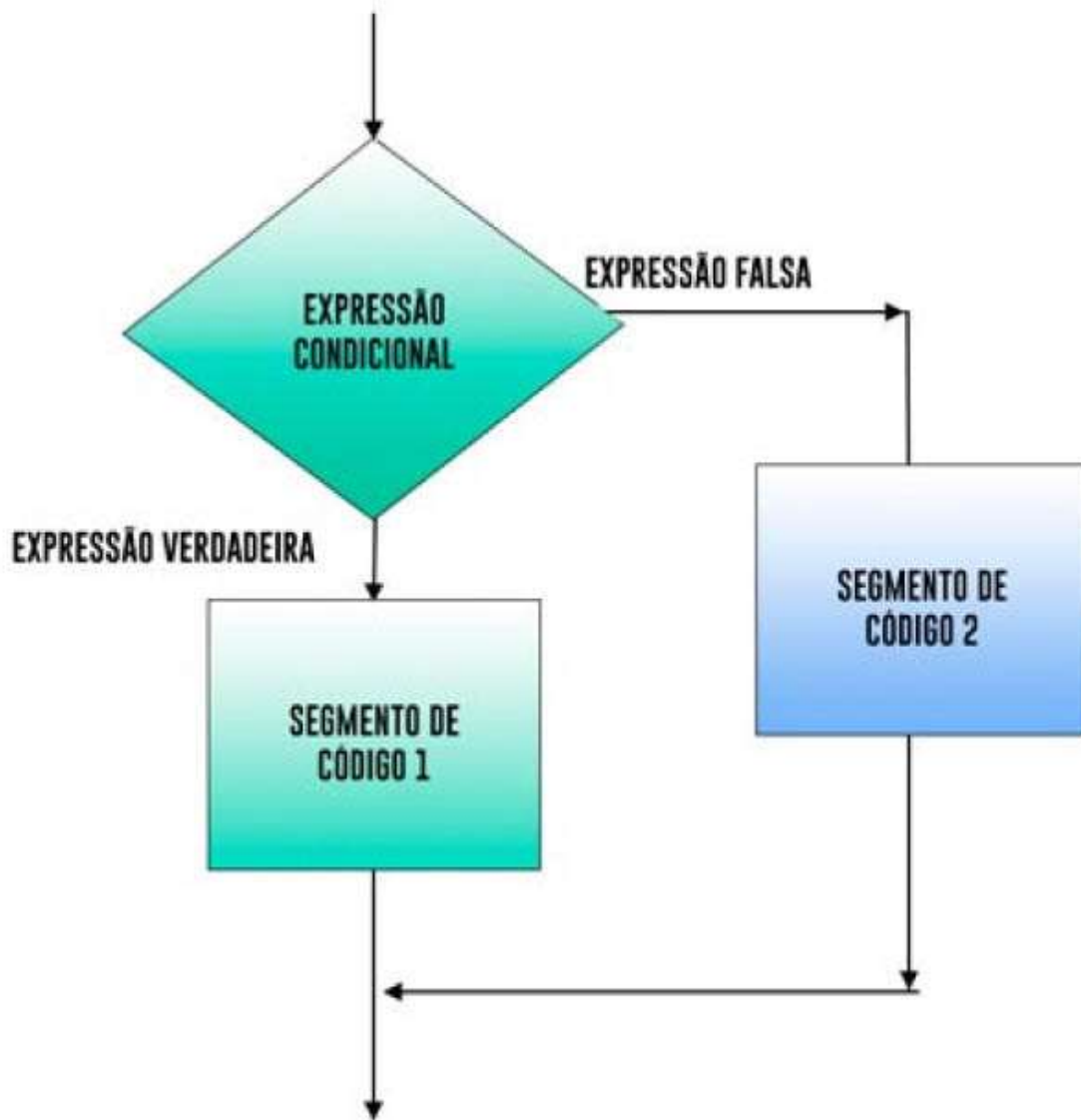
    return 0;
}
```

Observe que, ao digitar uma nota maior que 5, a estrutura condicional da linha 8 é verdadeira e a mensagem de aprovação é exibida, se for menor que 5, a da linha 10.

Estrutura de decisão composta

Como vimos anteriormente, a estrutura de decisão simples permite que um bloco seja executado caso uma expressão lógica seja verdadeira.

Agora acrescentaremos um novo bloco, que será executado no caso de a condição ser falsa. Assim, estamos nos referindo à estrutura **SE-ENTÃO-SENÃO**, conforme exibido na imagem a seguir.



A principal diferença deste tipo de estrutura condicional e a anterior reside no bloco marcado em azul na imagem anterior. Neste caso, quando a expressão condicional for avaliada como falsa, o bloco azul será executado.

Dessa forma, quando uma expressão condicional é avaliada, caso o seu valor seja verdadeiro, um bloco de comandos 1 é executado, caso contrário, será executado o bloco de comandos 2.

Após a execução do bloco de comandos selecionado pela expressão, é dada continuidade à execução da aplicação.

Na tabela a seguir, do lado esquerdo é exibido este código representado em PORTUGOL, e do lado direito o mesmo código, usando a Linguagem C. A principal diferença perante o caso anterior está relacionada nas linhas marcadas em amarelo, que representam o quadrado azul do diagrama. Assim,

caso a expressão condicional seja falsa, o código que será executado será o presente na estrutura *else*.

PORTUGOL	LINGUAGEM C
SE EXPRESSÃO_CONDICIONAL	if(expressao_condicional)
ENTÃO	{
BLOCO DE COMANDOS 1	bloco_de_comandos_1;
FIM_ENTÃO	}
	else
SENÃO	{
BLOCO DE COMANDOS 2	bloco_de_comandos_2;
FIM_SENÃO	}
FIM_SE	

Elaborada por Anderson Fernandes Pereira dos Santos.

Atenção!

A mesma regra com relação à quantidade de instruções e às chaves é utilizada neste caso. Portanto, se o bloco possuir apenas um comando, as chaves são opcionais, porém normalmente utilizadas. No caso de haver mais de uma instrução, estas chaves tornam-se obrigatórias.