

AULA 03

CONDICIONAIS

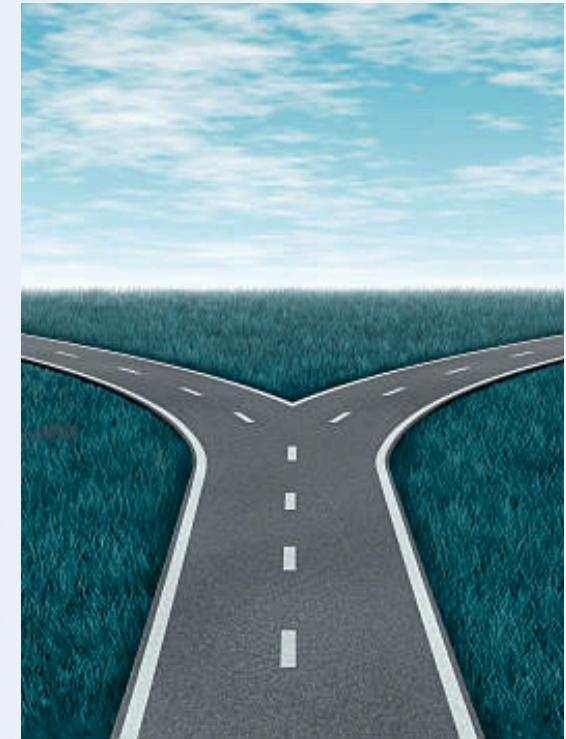
EDUBOT

O QUE SÃO CONDICIONAIS?

Condicionais são estruturas que ajudam o computador a tomar decisões. Elas dizem ao programa o que fazer se uma condição for verdadeira ou falsa.

Por exemplo:

Imagine que você está dirigindo e precisa atravessar da Asa Norte para Asa Sul porém o Eixão, via pela qual você geralmente vai, está fechado. Assim, você vai pela W3. Nessa situação, a condição para decidir como seria feito o trajeto era o Eixão estar aberto ou fechado e, como ele estava fechado, foi tomada a decisão de seguir pela W3



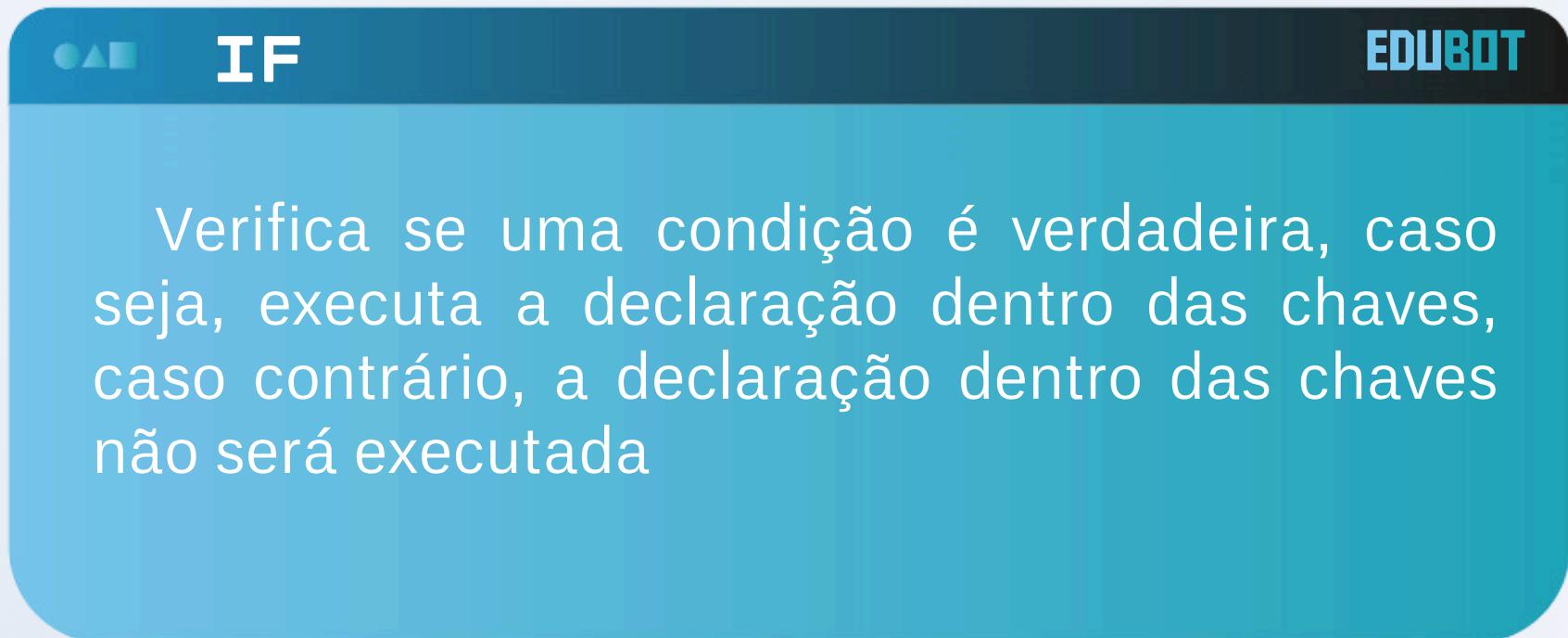
OPERADORES LÓGICOS

SÍMBOLO	OPERAÇÃO	EXEMPLOS
<	MENOR QUE	3 < 2 (FALSE)
>	MAIOR QUE	3 > 2 (TRUE)
<=	MENOR IGUAL	3 <= 2 (FALSE)
>=	MAIOR IGUAL	3 >= 2 (TRUE)
==	IGUAL À	3 == 2 (FALSE)
!=	DIFERENTE DE	3 != 2 (TRUE)

OPERADORES ARITMÉTICOS

SÍMBOLO	OPERAÇÃO	EXEMPLOS
+	ADIÇÃO	$4 + 8 = 12$
-	SUBTRAÇÃO	$8 - 4 = 4$
/	DIVISÃO	$16 / 2 = 8$
*	MULTIPLICAÇÃO	$8 * 2 = 16$
**	POTENCIAÇÃO	$2 ** 3 = 8$
%	MÓDULO (RESTO)	$8 \% 3 = 2$

ESTRUTURAS DE CONTROLE IF, ELSE E ELSE IF



The diagram illustrates the structure of an IF block. It features a blue rounded rectangle with a black header bar. On the left side of the header bar is the OAD logo, which consists of three white squares of increasing size from left to right. To the right of the logo is the word "IF" in white capital letters. On the far right of the header bar is the word "EDUBOT" in white capital letters. The main body of the block is light blue and contains the following text in white: "Verifica se uma condição é verdadeira, caso seja, executa a declaração dentro das chaves, caso contrário, a declaração dentro das chaves não será executada".

IF

EDUBOT

Verifica se uma condição é verdadeira, caso seja, executa a declaração dentro das chaves, caso contrário, a declaração dentro das chaves não será executada

EXEMPLO

AQUI UTILIZAREMOS “SE” PARA REPRESENTAR “IF”

Algoritmo para lavar louça:

Vá até a pia

Pegue um prato

Se o prato estiver sujo {

Lave o prato

}

EXEMPLO:

```
1 #include <Sparki.h> // inclui a biblioteca sparki
2
3 void setup()
4 {
5 }
6
7 void loop()
8 {
9     int cm = sparki.ping(); // mede a distancia pelo sensor do sparki
10
11    if (cm < 10) // se a distancia for menor que 10 centimetros
12    {
13        sparki.RGB(RGB_RED); // mostra a cor vermelha
14    }
15 }
```

ESTRUTURAS DE CONTROLE IF, ELSE E ELSE IF



ELSE

EDUBDT

A estrutura “ELSE” é utilizada caso a condição contida no “IF” seja falsa, orientando o programa o que fazer quando nesses casos, assim executando a declaração contidas nas chaves do “ELSE”.

EXEMPLO

AQUI UTILIZAREMOS “SE” PARA REPRESENTAR “IF”
E “CASO CONTRÁRIO” PARA “ELSE”

Algoritmo para lavar louça:

Vá até a pia
Pegue um prato
Se o prato estiver sujo {
 Lave o prato
}
Caso contrário{
 Guarda o prato no armario
}

EXEMPLO:

```
1 #include <Sparki.h> // inclui a biblioteca sparki
2
3 void setup()
4 {
5 }
6
7 void loop()
8 {
9     int cm = sparki.ping(); // mede a distancia pelo sensor do sparki
10
11    if (cm < 10) // se a distancia for menor que 10 centimetros
12    {
13        sparki.RGB(RGB_RED); // mostra a cor vermelha
14    }
15
16    else // caso não for menor que 10 cm, ou seja, maior ou igual a 10
17    {
18        sparki.RGB(RGB_BLUE); // mostra a cor azul
19    }
20 }
```

ESTRUTURAS DE CONTROLE IF, ELSE E ELSE IF



ELSE IF

EDUBDT

As estruturas “IF” e “ELSE” podem ser combinadas, dessa forma a estrutura “ELSE IF” verifica se o if anterior é falso, caso seja, verifica a condição imposta em “ELSE IF”, se verdadeira, executa a declaração entre chaves.

EXEMPLO

AQUI UTILIZAREMOS “SE” PARA REPRESENTAR “IF”
E “CASO CONTRÁRIO” PARA “ELSE”

Algoritmo para lavar louça:

```
Vá até a pia  
Pegue um prato  
Se o prato estiver sujo {  
    Lave o prato  
}  
Caso contrário se o prato estiver molhado{  
    Seque o prato  
}  
Caso contrário{  
    Guarde o prato no armario  
}
```

EXEMPLO

EM UMA NOVA SITUAÇÃO, EXISTE UMA PILHA DE BLUSAS DE CORES SORTIDAS. SUA MISSÃO É COLOCAR AS BLUSAS AZUIS EM UMA CAIXA E COLOCAR AS BLUSAS VERMELHAS NO ARMÁRIO. AS BLUSAS DE OUTRAS CORES DEVEM SER DEIXADAS NA PILHA. A PALAVRA “SE” REPRESENTA O “IF”, A EXPRESSÃO “CASO CONTRÁRIO” REPRESENTA O “ELSE”:

EXEMPLO

AQUI UTILIZAREMOS “SE” PARA REPRESENTAR “IF”
E “CASO CONTRÁRIO” PARA “ELSE” :

Algoritmo para separar blusas vermelhas e azuis:

Pegue uma blusa

Se a blusa for azul{

Coloque a blusa na caixa

}

Caso contrário se a blusa for vermelha{

Coloque a blusa no armário

{

Caso contrário {

Devolva a blusa para a pilha

}

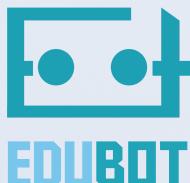
EXEMPLO :

```
1 #include <Sparki.h> // inclui a biblioteca sparki
2
3 void setup()
4 {
5 }
6
7 void loop()
8 {
9     int cm = sparki.ping(); // mede a distancia pelo sensor do sparki
10
11    if (cm < 10) // se a distancia for menor que 10 centimetros
12    {
13        sparki.RGB(RGB_RED); // mostra a cor vermelha
14    }
15
16    else if (cm == 10) // se a distancia for igual que 10 centimetros
17    {
18        sparki.RGB(RGB_GREEN); // mostra a cor verde
19    }
20
21    else // caso não for menor ou igual a 10, ou seja, maior que 10
22    {
23        sparki.RGB(RGB_BLUE); // mostra a cor azul
24    }
25 }
```

ATIVIDADE 1

DESENVOLVA UM PROGRAMA QUE LÊ OS VALORES DOS SENSORES DE LINHA (ESQUERDO, CENTRAL E DIREITO) E REALIZA DIFERENTES MOVIMENTOS COM BASE NESSES VALORES.

SE A LEITURA DO SENSOR ESQUERDO FOR MENOR QUE 500, O ROBÔ DEVE ANDAR PARA FRENTE. SE A LEITURA DO SENSOR CENTRAL FOR MAIOR QUE 500, O ROBÔ DEVE ANDAR PARA TRÁS. SE A LEITURA DO SENSOR DIREITO FOR IGUAL A 500, O ROBÔ DEVE GIRAR 90 GRAUS. ADICIONE UM PEQUENO ATRASO DE 100 MILISSEGUNDOS ENTRE AS LEITURAS PARA EVITAR LEITURAS MUITO RÁPIDAS.



RESPOSTA:

```
#include <Sparki.h> // Inclui a biblioteca Sparki

void setup() {
    // Deixe o setup vazio conforme solicitado
}

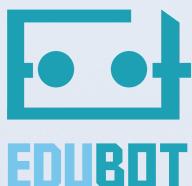
void loop() {
    // Lê os valores dos sensores de Linha
    int leituraEsquerda = sparki.lineLeft();
    int leituraCentral = sparki.lineCenter();
    int leituraDireita = sparki.lineRight();

    // Verificando a Leitura do sensor esquerdo
    if (leituraEsquerda < 500) {
        sparki.moveForward(); // Move para frente
    }
    // Verificando a Leitura do sensor central
    else if (leituraCentral > 500) {
        sparki.moveBackward(); // Move para trás
    }
    // Verificando a Leitura do sensor direito
    else if (leituraDireita == 500) {
        sparki.moveRight(); // Gira 90 graus para a direita
    }

    // Espera um curto período antes da próxima Leitura
    delay(100); // Delay de 100 milissegundos
}
```

ATIVIDADE 2

DESENVOLVA UM PROGRAMA QUE LÊ OS VALORES DOS SENSORES DE LINHA (ESQUERDO, CENTRAL E DIREITO) E ACIONA DIFERENTES LUZES RGB COM BASE NESSES VALORES. SE A LEITURA DO SENSOR ESQUERDO FOR MENOR QUE 500, LIGUE A LUZ VERMELHA. SE A LEITURA DO SENSOR CENTRAL FOR MENOR QUE 500, LIGUE A LUZ VERDE. SE A LEITURA DO SENSOR DIREITO FOR MENOR QUE 500, LIGUE A LUZ AZUL. CASO CONTRÁRIO, DESLIGUE TODAS AS LUZES. ADICIONE UM PEQUENO ATRASO DE 100 MILISSEGUNDOS ENTRE AS LEITURAS PARA EVITAR LEITURAS MUITO RÁPIDAS.



RESPOSTA:

```
#include <Sparki.h> // Inclui a biblioteca Sparki

void setup() {
    // Deixe o setup vazio nessa atividade
}

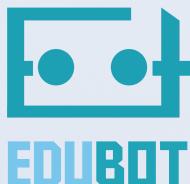
void loop() {
    // Lê os valores dos sensores de Linha
    int leituraEsquerda = sparki.lineLeft();
    int leituraCentral = sparki.lineCenter();
    int leituraDireita = sparki.lineRight();

    // Verificando a Leitura do sensor esquerdo
    if (leituraEsquerda < 500) {
        sparki.RGB(RGB_RED); // Liga a Luz vermelha
    }
    else if (leituraCentral < 500) {
        sparki.RGB(RGB_GREEN); // Liga a Luz verde
    }
    else if (leituraDireita < 500) {
        sparki.RGB(RGB_BLUE); // Liga a Luz azul
    }
    else {
        sparki.RGB(RGB_OFF); // Desliga todas as Luzes
    }

    // Espera um curto período antes da próxima leitura
    delay(100); // Delay de 100 milissegundos
}
```

ATIVIDADE 3

DESENVOLVA UM PROGRAMA QUE LÊ O VALOR DO SENSOR DE DISTÂNCIA E REALIZA DIFERENTES MOVIMENTOS COM BASE NESSE VALOR. SE A LEITURA DO SENSOR DE DISTÂNCIA FOR MENOR QUE 20 CM, O ROBÔ DEVE ANDAR PARA FRENTE. SE A LEITURA DO SENSOR DE DISTÂNCIA FOR MAIOR QUE 50 CM, O ROBÔ DEVE ANDAR PARA TRÁS. SE A LEITURA DO SENSOR DE DISTÂNCIA ESTIVER ENTRE 20 CM E 50 CM (INCLUSIVE), O ROBÔ DEVE GIRAR 90 GRAUS. ADICIONE UM PEQUENO ATRASO DE 100 MILISSEGUNDOS ENTRE AS LEITURAS PARA EVITAR LEITURAS MUITO RÁPIDAS



RESPOSTA:

```
#include <Sparki.h> // Inclui a biblioteca Sparki

void setup() {
    // Deixe o setup vazio
}

void loop() {
    // Lê o valor do sensor de distância em centímetros
    int distancia = sparki.ping();

    // Verificando a leitura do sensor de distância
    if (distancia < 20) {
        sparki.moveForward(); // Move para frente
    }
    else if (distancia > 50) {
        sparki.moveBackward(); // Move para trás
    }
    else {
        sparki.moveRight(); // Gira 90 graus para a direita
    }

    // Espera um curto período antes da próxima leitura
    delay(100); // Delay de 100 milissegundos
}
```