



Robótica Educacional

[PÁGINA INICIAL](#)[MATERIAL DIDÁTICO](#)[PROJETOS](#)[COMPONENTES](#)[EQUIPE](#)[CONTATO](#)[DOCUMENTOS](#)[Estruturas de Controle](#)[Estruturas de Seleção](#)[Seleção Simples](#)[Seleção Composta](#)[Seleção Encadeada](#)

Estruturas de Controle

Os conceitos de tipos de dados, operadores, variáveis, constantes, atribuições, entradas e saídas de dados, bem como os comandos de uma determinada linguagem, representam um conjunto de ações.

Para que esse conjunto de ações se torne viável, deve existir uma perfeita relação lógica intrínseca ao modo pelo qual essas ações são executadas.

Para executar as ações é necessário utilizar as estruturas básicas de controle de fluxo de execução - sequencial, seleção e repetição. Com a combinação dessas estruturas, poderemos criar algoritmos para solucionar os problemas existentes em um processamento.

As rotinas programáveis são organizadas em: entrada de dados, processamento de dados e saída de dados:

- Entrada de dados

O Arduino recebe os dados fornecidos por uma pessoa ou por um dispositivo e processa através de instruções específicas de programação.

- Processamento de dados

Após a entrada de dados, segue-se para a etapa de processamento de dados. Nesta etapa, o Arduino é capaz de processar esses dados em novos e/ou informações.

- Saída de dados

Após o processamento dos dados é possível apresentar os dados processados e/ou transformados.

Iremos desenvolver algumas aplicações utilizando o Arduino. Ao final de cada aplicação, espera-se que você consiga identificar a estrutura apresentada, compreender as ações de cada componente utilizado, bem como associar o código do programa com o algoritmo da aplicação.

Estruturas de Seleção

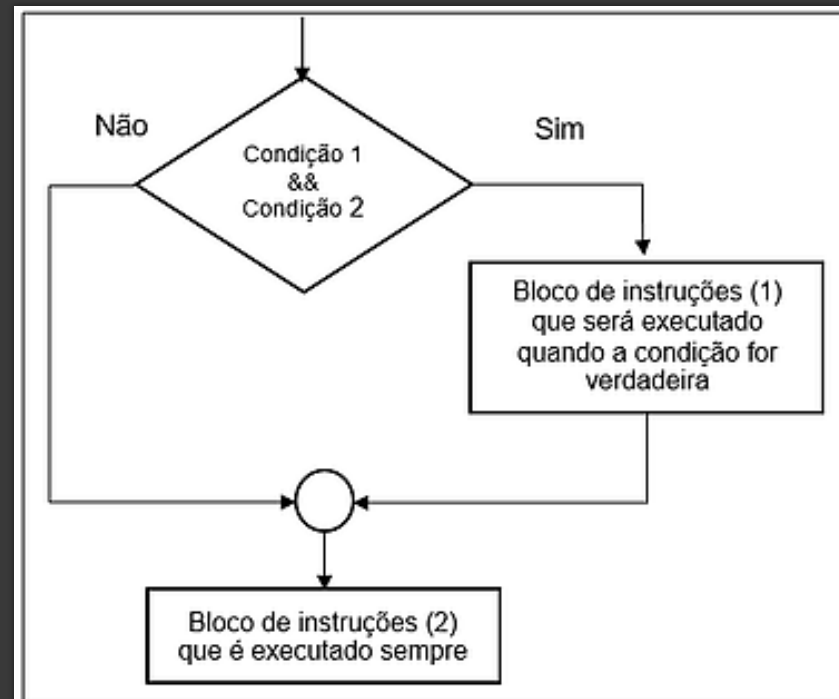
Através da Estrutura de Seleção, quando uma determinada condição for satisfeita ou não, é possível determinar qual grupo de instruções será executado. Essas condições são representadas por operadores lógicos e relacionais.

Com a Estrutura de Seleção o algoritmo poderá tomar decisões, buscando atender os critérios determinados.

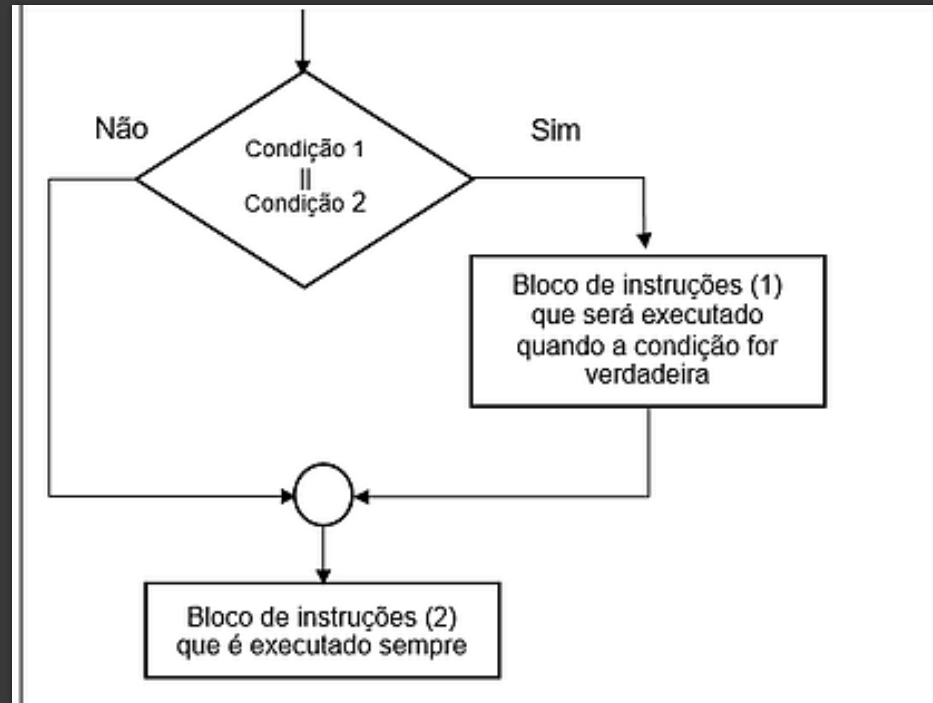
Os símbolos do diagrama de blocos utilizados são de decisão e conexão.

A representação dos operadores lógicos através do diagrama de blocos, tem a seguinte estrutura:

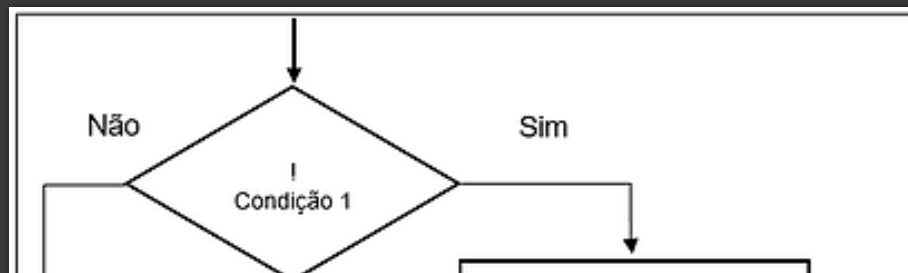
Quadro MD8 - Operador Lógico de Conjunção (&&)

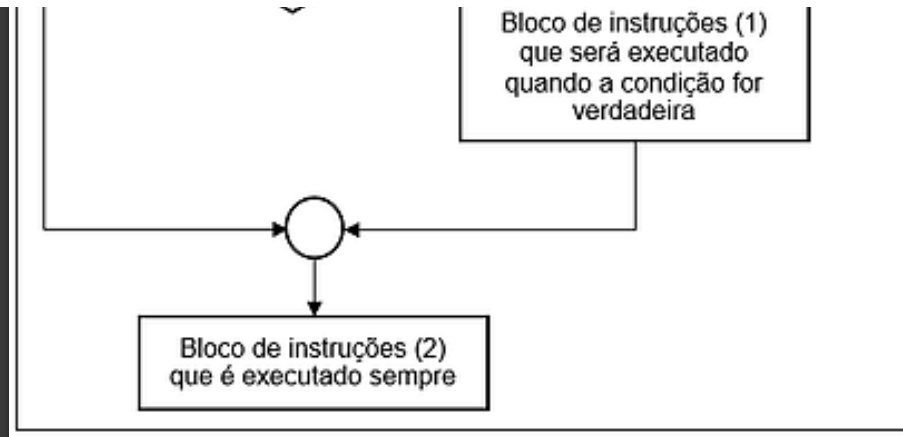


Quadro MD9 - Operador Lógico de Disjunção (II)



Quadro MD10 - Operador Lógico de Negação (!)





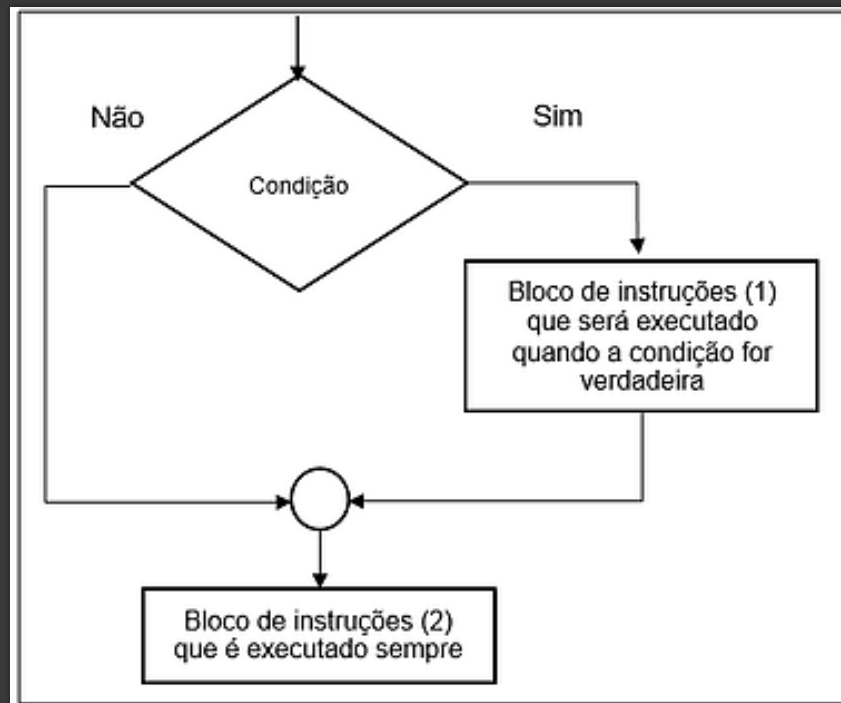
Seleção Simples

A estrutura de Seleção Simples (Quadro MD11) é utilizada quando for necessário verificar uma condição antes de executar um bloco de instruções.

Um bloco de instruções é executado quando uma condição for verdadeira, caso contrário o bloco será ignorado.

No diagrama de blocos a palavra **Sim** é utilizada com o objetivo de mostrar qual é o bloco de instruções que será executado quando a condição for verdadeira. A palavra **Não** representa a condição contrária. Quando uma condição for falsa nenhum bloco de instruções será executado.

Quadro MD11 - Estrutura de Seleção Simples



Desenvolver um projeto utilizando três botões, onde cada botão acenderá um LED específico. Quando os três LEDs estiverem acesos o programa deverá disparar um alarme.

Projeto - 01

Componentes

Qtd - Descrição

01 - Arduino Mega 2560

Instruções

Push-button nas portas digitais (2, 3 e 4)

LEDs nas portas digitais (10, 11 e 12)

01 - Arduino Mega 2560

01 - Protoboard

03 - Push-Button

01 - Buzzer

03 - LEDs (01 Vermelho – 01 Amarelo – 01 Verde)

03 - Resistores 10K Ω para os Push-Button

03 - Resistores 220 Ω para os Leds

Jumpers

Buzzer na porta digital (8)

Conhecimentos necessários

- Arduino IDE

- Variáveis e Constantes

- Operadores Relacionais

- Operadores Lógicos

- Portas Digitais

- Construção de Algoritmos: Estrutura de Seleção

Simple

- Especificações dos seguintes componentes:

(Protoboard) (Push-Button) (Buzzer) (Led) (Resistor)

(Jumper)

Observação:

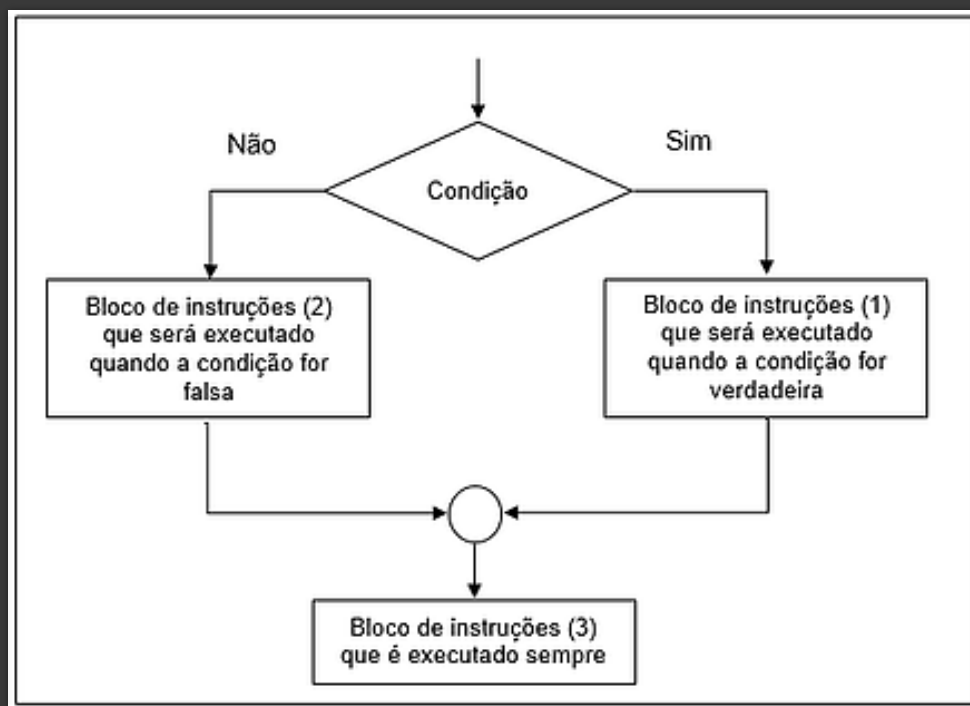
Os resistores serão utilizados como limitadores de corrente.

Seleção Composta

A estrutura de Seleção Composta é utilizada quando for necessário executar 2 (dois) blocos de instruções, um para a condição verdadeira e o outro para a condição falsa.

O bloco de instruções (1) é executado quando a condição for verdadeira, caso contrário, ou seja, a condição for falsa, o bloco de instruções (2) será executado.

Quadro MD12 - Estrutura de Seleção Composta



Para exemplificar, iremos resolver a seguinte situação:

Desenvolver um projeto utilizando três botões, onde cada botão acenderá um LED específico. Quando os três LEDs estiverem acesos o programa deverá disparar um alarme.

Componentes

Qtd - Descrição

01 - Arduino Mega 2560

01 - Protoboard

03 - Push-Button

01 - Buzzer

03 - LEDs (01 Vermelho – 01 Amarelo – 01 Verde)

03 - Resistores 10K Ω para os Push-Button

03 - Resistores 220 Ω para os Leds

Jumpers

Instruções

Push-button nas portas digitais (2, 3 e 4)

LEDs nas portas digitais (10, 11 e 12)

Buzzer na porta digital (8)

Conhecimentos necessários

- Arduino IDE
- Variáveis e Constantes
- Operadores Relacionais
- Operadores Lógicos
- Portas Digitais
- Construção de Algoritmos: Estrutura de Seleção Composta

Observação:

Os resistores serão utilizados como limitadores de corrente.

O projeto anterior será utilizado neste exemplo. A mudança ocorrerá somente no código.

- Especificações dos seguintes componentes:
(Protoboard) (Push-Button) (Buzzer) (Led) (Resistor)
(Jumper)

Seleção Encadeada

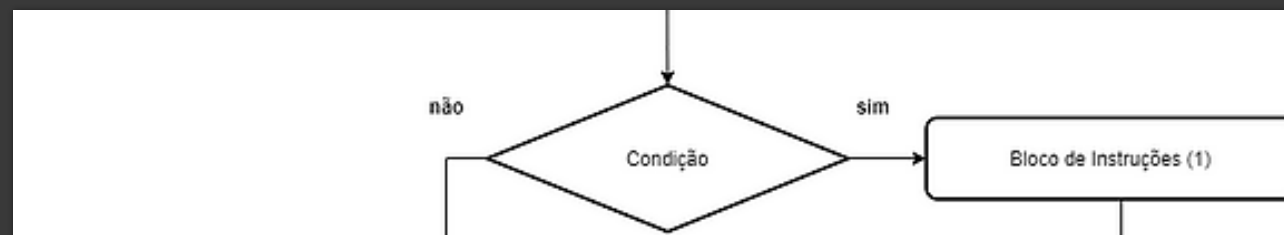
Quando for necessário executar bloco de instruções com várias possibilidades de situações, então deveremos utilizar a seleção encadeada.

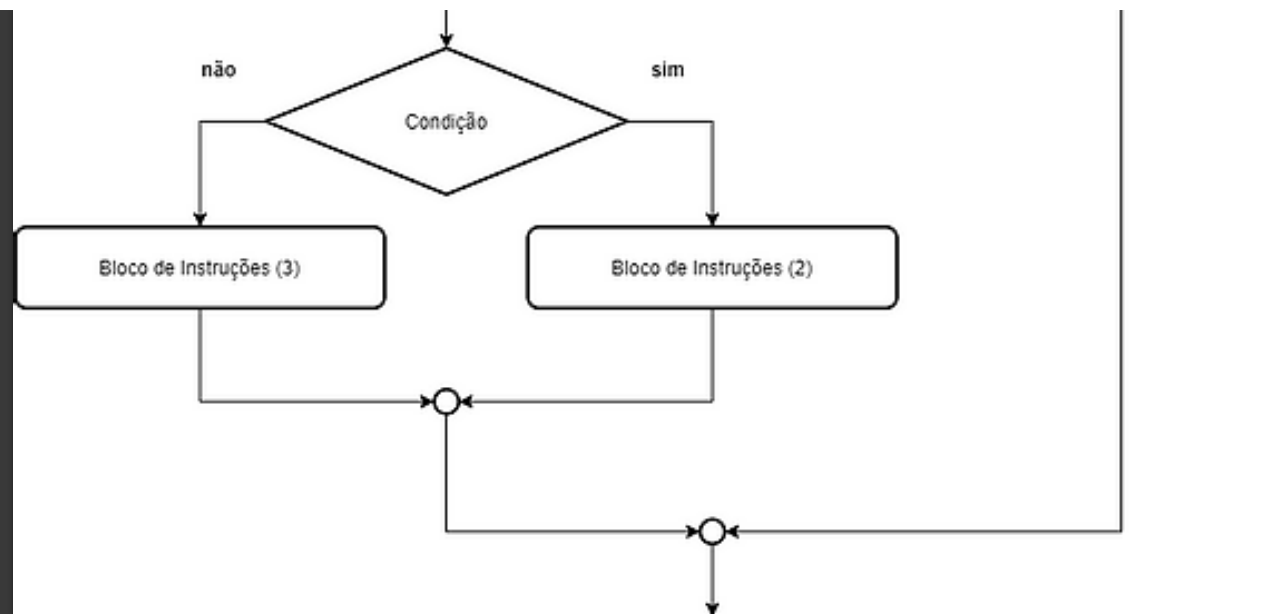
A seleção encadeada permite estrutura de seleção dentro de uma outra estrutura de seleção.

O bloco de instrução (1) é executado quando um conjunto de anterior de instruções for satisfeito. Caso contrário, são necessárias outras condições.

Chamamos de condição encadeada pela possibilidade de realizar a verificação através de n estruturas.

Quadro MD13 - Estrutura de Seleção Encadeada





Para exemplificar, iremos resolver a seguinte situação:

Desenvolver um projeto utilizando três botões, onde cada botão acenderá um LED específico. Quando dois LEDs estiverem acesos o programa deverá disparar um alarme.

Projeto - 03

Componentes

Qtd - Descrição

01 - Arduino Mega 2560

01 - Protoboard

Instruções

Push-button nas portas digitais (2, 3 e 4)

LEDs nas portas digitais (10, 11 e 12)

Buzzer na porta digital (8)

03 - Push-Button
01 - Buzzer
03 - LEDs (01 Vermelho – 01 Amarelo – 01 Verde)
03 - Resistores 10K Ω para os Push-Button
03 - Resistores 220 Ω para os Leds
Jumpers

Conhecimentos necessários

- Arduino IDE
 - Variáveis e Constantes
 - Operadores Relacionais
 - Operadores Lógicos
 - Portas Digitais
 - Construção de Algoritmos: Estrutura de Seleção Encadeada
- Especificações dos seguintes componentes:
(Protoboard) (Push-Button) (Buzzer) (Led) (Resistor)
(Jumper)

Observação:

Os resistores serão utilizados como limitadores de corrente.

O projeto anterior será utilizado neste exemplo. A mudança ocorrerá somente no código.

Voltar ao Topo ^

CONTATOS



e-mail: euderfs@gmail.com

© 2021 by Euder Santos