**Programação trabalho em grupo arduino**

#include <VarSpeedServo.h>

#include <Arduino.h>

// Declaração dos servos

VarSpeedServo motor1; // Servo responsável pelo giro

VarSpeedServo motor2; // Servo responsável por subir e descer

VarSpeedServo motor3; // Servo responsável pelo avanço e recuo

VarSpeedServo motor4; // Servo responsável pela abertura/fechamento da garra

// Definição dos pinos do sensor de cor

const int S0 = 10;

const int S1 = 11;

const int S2 = 8;

const int S3 = 13;

const int sensorOut = 12;

// Variáveis para armazenar a frequência de saída

int redFrequency = 0;

int greenFrequency = 0;

int blueFrequency = 0;

// Função de configuração inicial

void setup() {

motor1.attach(3);

motor2.attach(5);

motor3.attach(6);

motor4.attach(9);

pinMode(S0, OUTPUT);

pinMode(S1, OUTPUT);

pinMode(S2, OUTPUT);

pinMode(S3, OUTPUT);

pinMode(sensorOut, INPUT);

digitalWrite(S0, HIGH);

digitalWrite(S1, LOW);

Serial.begin(9600);

}

// Função para movimentar o braço

void moverParaPosicao(VarSpeedServo &motor, int angulo, int velocidade = 60) {

motor.slowmove(angulo, velocidade);

delay(100);

}

// Função para ler a cor

void lerCor() {

// Leitura da frequência do vermelho

digitalWrite(S2, LOW);

digitalWrite(S3, LOW);

redFrequency = pulseIn(sensorOut, LOW);

// Leitura da frequência do verde

digitalWrite(S2, HIGH);

digitalWrite(S3, HIGH);

greenFrequency = pulseIn(sensorOut, LOW);

// Leitura da frequência do azul

digitalWrite(S2, LOW);

digitalWrite(S3, HIGH);

blueFrequency = pulseIn(sensorOut, LOW);

// Impressão dos valores no Serial Monitor

Serial.print("Red: ");

Serial.print(redFrequency);

Serial.print(" - Green: ");

Serial.print(greenFrequency);

Serial.print(" - Blue: ");

Serial.println(blueFrequency);

}

// Função para executar o ciclo para objetos vermelhos

void executarCicloVermelho() {

// Passo 4: Garra fechada

moverParaPosicao(motor4, 175);

delay(1000);

// Passo 5: Garra fechada braço para cima

moverParaPosicao(motor4, 175);

moverParaPosicao(motor1, 0);

moverParaPosicao(motor2, 152);

moverParaPosicao(motor3, 126);

delay(1000);

// Aproximar até a primeira caixa

moverParaPosicao(motor4, 175);

moverParaPosicao(motor1, 90);

moverParaPosicao(motor2, 152);

moverParaPosicao(motor3, 126);

delay(1000);

// Passo 6: Movimentar até a primeira caixa

moverParaPosicao(motor4, 175);

moverParaPosicao(motor1, 90);

moverParaPosicao(motor2, 175);

moverParaPosicao(motor3, 140);

delay(1000);

// Passo 7: Abrir a garra

moverParaPosicao(motor4, 30); // Abre garra

delay(1000);

// Passo 8: Voltar a posição inicial

moverParaPosicao(motor4, 175);

moverParaPosicao(motor1, 0);

moverParaPosicao(motor2, 175);

moverParaPosicao(motor3, 100);

delay(1000);

// Passo 2: Abrir a garra

moverParaPosicao(motor4, 30);

delay(1000);

// Passo 3: Pegar a peça

moverParaPosicao(motor4, 30 , 100);

moverParaPosicao(motor1, 0 , 100);

moverParaPosicao(motor2, 151 , 100);

moverParaPosicao(motor3, 172 , 100);

delay(1000);

}

// Função para executar o ciclo para objetos azuis

void executarCicloAzul() {

// Passo 4: Garra fechada

moverParaPosicao(motor4, 175);

delay(2000);

// Passo 5: Garra fechada braço para cima

moverParaPosicao(motor4, 175);

moverParaPosicao(motor1, 0);

moverParaPosicao(motor2, 152);

moverParaPosicao(motor3, 126);

delay(1000);

// Passo 6: Movimentar até a segunda caixa

moverParaPosicao(motor4, 175);

moverParaPosicao(motor1, 150);

moverParaPosicao(motor2, 175);

moverParaPosicao(motor3, 100);

delay(1000);

// Passo 7: Movimentar até a segunda caixa e depositar

moverParaPosicao(motor4, 175);

moverParaPosicao(motor1, 150);

moverParaPosicao(motor2, 175);

moverParaPosicao(motor3, 140);

delay(1000);

// Passo 8: Abrir a garra

moverParaPosicao(motor4, 30);

delay(1000);

// Passo 9: Fecha a garra

moverParaPosicao(motor4, 175);

delay(1000);

// Passo 10: Retorna a posição inicial

moverParaPosicao(motor4, 175);

moverParaPosicao(motor1, 0);

moverParaPosicao(motor2, 175);

moverParaPosicao(motor3, 100);

delay(1000);

// Passo 2: Abrir a garra

moverParaPosicao(motor4, 30);

delay(1000);

// Passo 3: Pegar a peça

moverParaPosicao(motor4, 30);

moverParaPosicao(motor1, 0);

moverParaPosicao(motor2, 151);

moverParaPosicao(motor3, 172);

delay(1000);

}

void loop() {

// Inicialmente, mover para a posição de leitura

moverParaPosicao(motor4, 30);

moverParaPosicao(motor1, 0);

moverParaPosicao(motor2, 151);

moverParaPosicao(motor3, 172); // Abrir garra

delay(2000);

lerCor(); // Ler a cor após abrir a garra

// Decisão baseada na cor detectada

if (redFrequency < greenFrequency && redFrequency < blueFrequency && redFrequency < 200) {

executarCicloVermelho();

} else if (blueFrequency < redFrequency && blueFrequency < greenFrequency && blueFrequency < 200) {

executarCicloAzul();

} else {

// Ficar em posição de leitura

moverParaPosicao(motor4, 30);

moverParaPosicao(motor1, 0);

moverParaPosicao(motor2, 151);

moverParaPosicao(motor3, 172); // Abrir garra

delay(2000);

}

}