

Código

Caída Libre

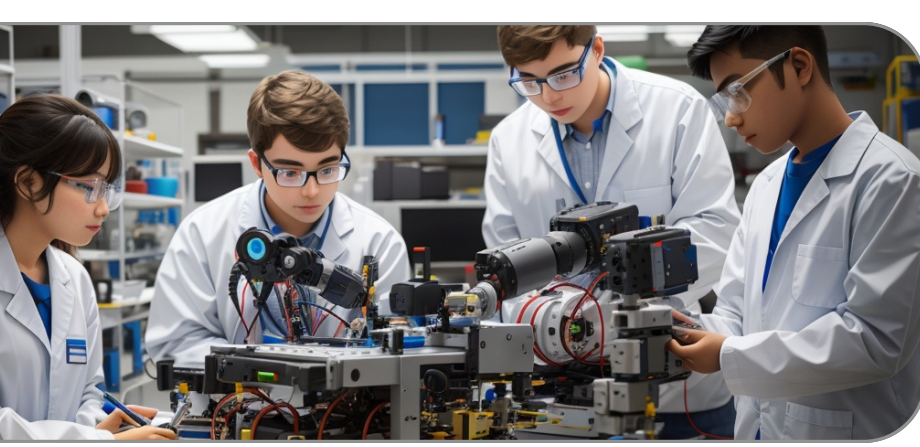
```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Servo.h>
#include <SoftwareSerial.h>

// Definición de pines para los sensores y los servos
const int sensorPin1 = 2;
const int sensorPin2 = 3;
const int servoPin1 = 9;
const int sensorPin3 = 4;
const int sensorPin4 = 5;
const int servoPin2 = 10;
const int sensorPin5 = 6;
const int sensorPin6 = 7;
const int servoPin3 = 11;

LiquidCrystal_I2C lcd(0x27, 16, 2);
Servo servo1;
Servo servo2;
Servo servo3;

SoftwareSerial BTSerial(8, 9); // RX | TX

long startTime;
long stopTime;
long micro;
float seg;
float vel;
```



Código

Caída Libre

```
bool tiempoCumplido1 = false;  
bool tiempoCumplido2 = false;  
bool tiempoCumplido3 = false;  
bool giroRealizado1 = false;  
bool giroRealizado2 = false;  
bool giroRealizado3 = false;
```

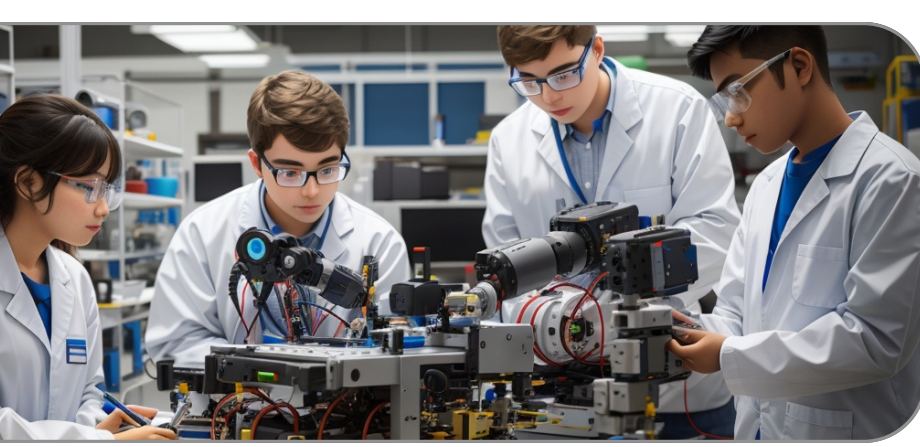
```
void setup() {  
  lcd.init();  
  lcd.backlight();  
  Serial.begin(9600);  
  BTSerial.begin(9600);
```

```
  pinMode(sensorPin1, INPUT);  
  pinMode(sensorPin2, INPUT);  
  pinMode(sensorPin3, INPUT);  
  pinMode(sensorPin4, INPUT);  
  pinMode(sensorPin5, INPUT);  
  pinMode(sensorPin6, INPUT);
```

```
  servo1.attach(servoPin1);  
  servo2.attach(servoPin2);  
  servo3.attach(servoPin3);
```

```
  servo1.write(0);  
  servo2.write(0);  
  servo3.write(0);
```

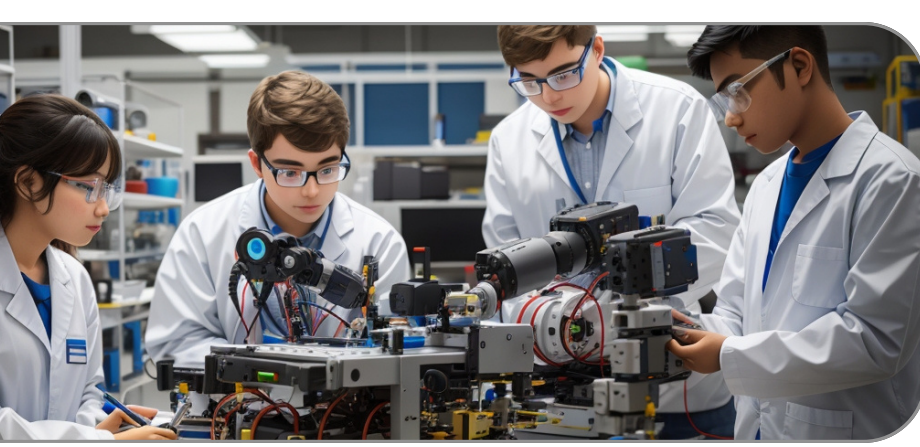
```
}
```



Código

Caída Libre

```
void loop() {  
  if (BTSerial.available()) {  
    char command = BTSerial.read();  
  
    switch (command) {  
      case 'A':  
        operateServo(sensorPin1, sensorPin2, servo1, tiempoCumplido1, giroRealizado1);  
        break;  
      case 'B':  
        operateServo(sensorPin3, sensorPin4, servo2, tiempoCumplido2, giroRealizado2);  
        break;  
      case 'C':  
        operateServo(sensorPin5, sensorPin6, servo3, tiempoCumplido3, giroRealizado3);  
        break;  
      default:  
        break;  
    }  
  }  
}  
  
void operateServo(int sensorPinA, int sensorPinB, Servo &servo, bool &tiempoCumplido,  
bool &giroRealizado) {  
  while (digitalRead(sensorPinA) == HIGH || digitalRead(sensorPinB) == HIGH) {}  
  
  pinMode(sensorPinA, INPUT);  
  pinMode(sensorPinB, INPUT);  
  
  while (!tiempoCumplido) {  
    if (!giroRealizado) {  
      servo.write(90);  
      delay(1000);  
      servo.write(0);  
      giroRealizado = true;  
    }  
  }  
}
```



Código

Caída Libre

```
stopTime = micros();  
micro = stopTime - startTime;  
seg = micro * 0.000001;  
  
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print("Tiempo (s): ");  
lcd.print(seg, 4);  
  
vel = 2 * (0.97 / seg);  
lcd.setCursor(0, 1);  
lcd.print("Veloc (m/s): ");  
lcd.print(vel, 4);  
  
if (millis() > 5000) {  
    tiempoCumplido = true;  
}  
}  
  
delay(5000);  
}
```