3.1-Blink LED

void setup()

{

pinMode(13, OUTPUT);

}

void loop()

{

digitalWrite(13, HIGH);

delay(1000); // Wait for 1000 millisecond(s)

digitalWrite(13, LOW);

delay(1000); // Wait for 1000 millisecond(s)

}

3.2-Button Led

void setup()

{

pinMode(2, INPUT);

pinMode(13,OUTPUT);

}

void loop()

{

if(digitalRead(2)==1)

{

digitalWrite(13,HIGH);

}

else

{

digitalWrite(13,LOW);

}

}

3.3-LED with Digital Counter

const int numpins=4;

int ledpins[numpins]={2,3,4,5};

void setup(){

for(int i=0;i<numpins;i++){

pinMode(ledpins[i],OUTPUT);

}

}

void loop(){

for(int num=1;num<=15;num++){

for(int bit=0;bit<numpins;bit++){

digitalWrite(ledpins[bit],bitRead(num,bit));

}

delay(1000);

}

}

3.4-RGB

int redPin= 13;

int greenPin = 12;

int bluePin = 11;

void setup() {

//Defining the pins as OUTPUT

pinMode(redPin, OUTPUT);

pinMode(greenPin, OUTPUT);

pinMode(bluePin, OUTPUT);

}

void loop() {

setColor(255, 0, 0); // Red Color

delay(1000);

setColor(0, 255, 0); // Green Color

delay(1000);

setColor(0, 0, 255); // Blue Color

delay(1000);

}

void setColor(int redValue, int greenValue, int blueValue) {

analogWrite(redPin, redValue);

analogWrite(greenPin, greenValue);

analogWrite(bluePin, blueValue);

}

4.1-Potentiometer

int pot=A0;

void setup()

{

Serial.begin(9600);

}

void loop()

{

int potvalue=analogRead(pot);

Serial.print("potvalue");

Serial.println(potvalue);

delay(10);

}

4.2-photoresistor

void setup()

{

pinMode(A0, INPUT);

Serial.begin(9600);

}

void loop()

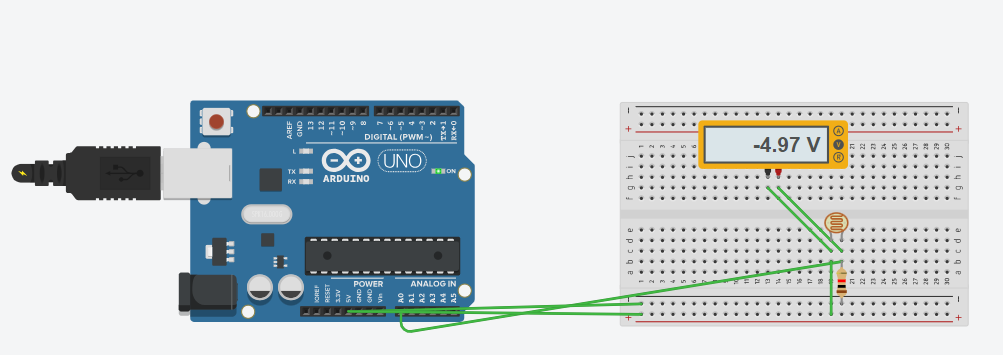
{

int lightvalue=analogRead(A0);

Serial.println(lightvalue);

delay(1000);

}



5.PIR sensor

void setup()

{

pinMode(2, INPUT);

pinMode(13, OUTPUT);

pinMode(9, OUTPUT);

}

void loop()

{

if (digitalRead(2) >= HIGH) {

digitalWrite(13, HIGH);

tone(9, 523, 1000); // play tone 60 (C5 = 523 Hz)

} else {

digitalWrite(13, LOW);

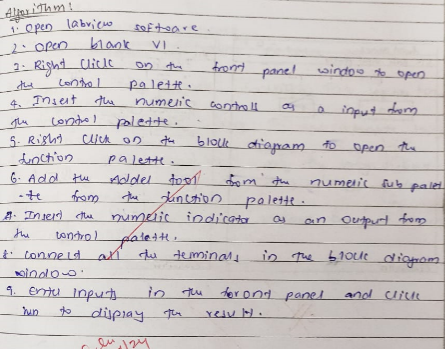
noTone(9);

}

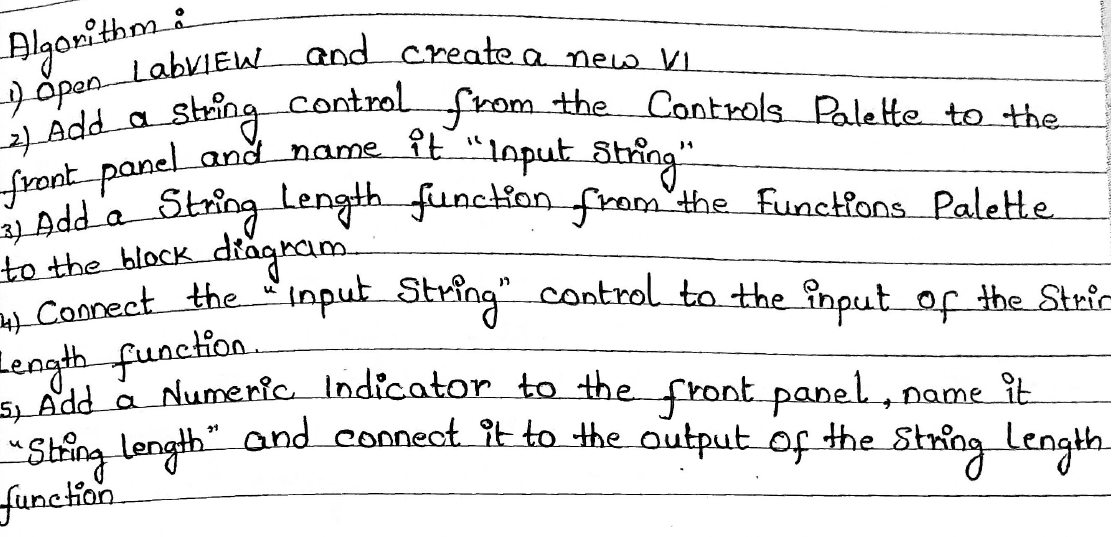
delay(1); // Wait for 1 millisecond(s)

}

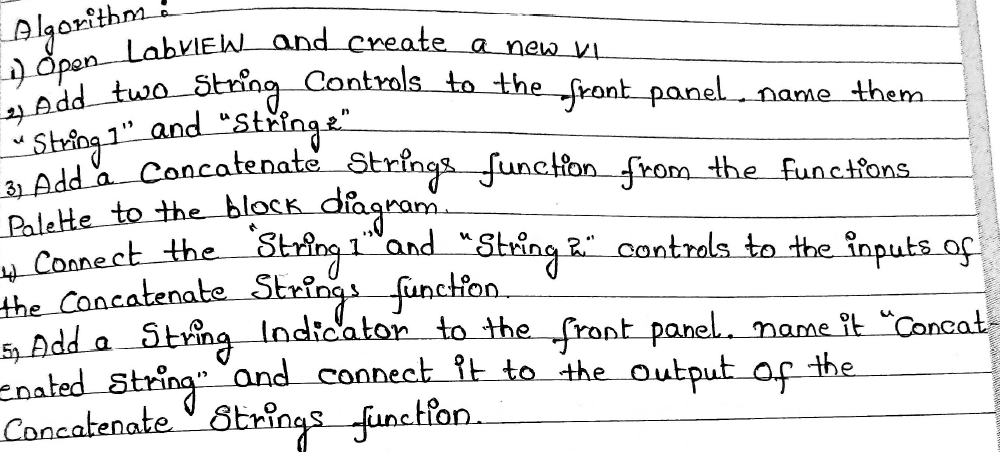
6-Addition



7A-str length



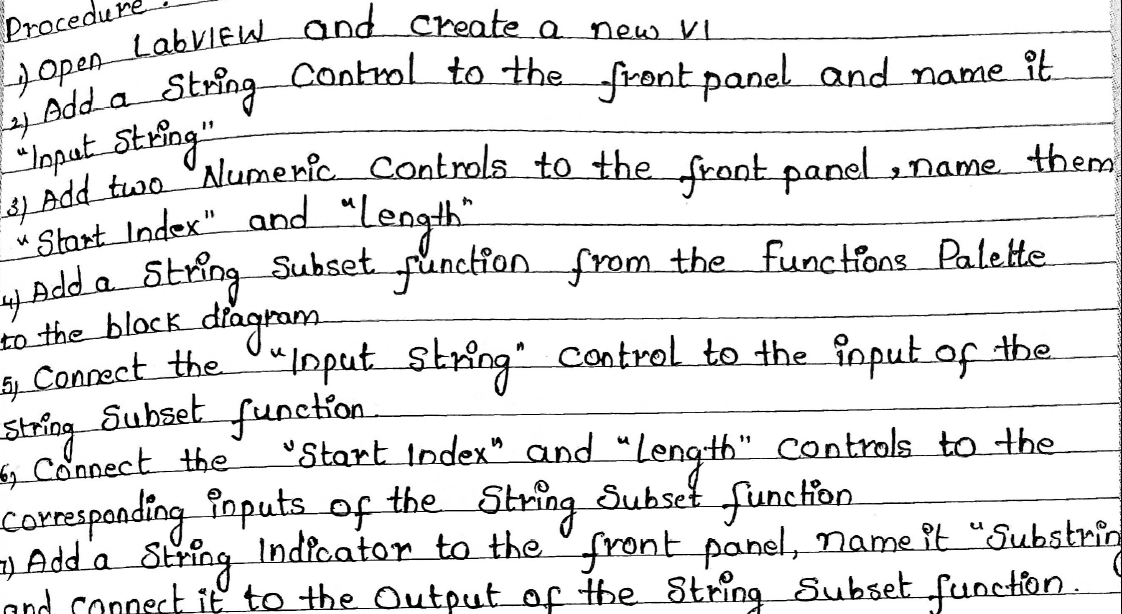
7**B-concat**

****

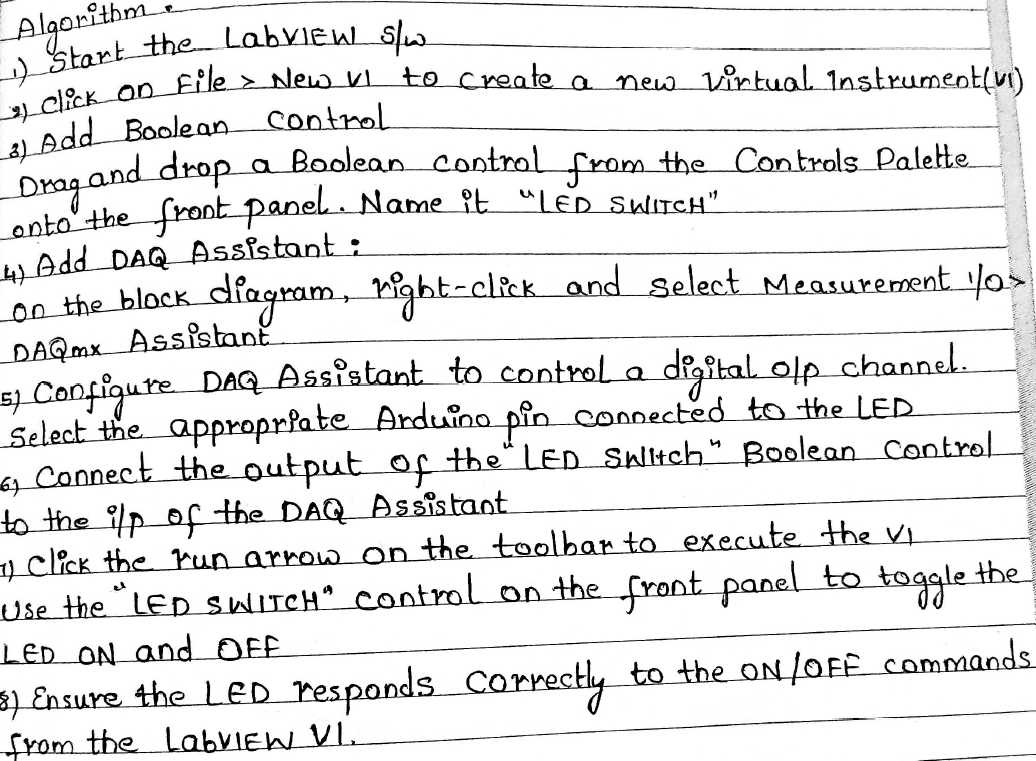
****

**7C-substring**

****

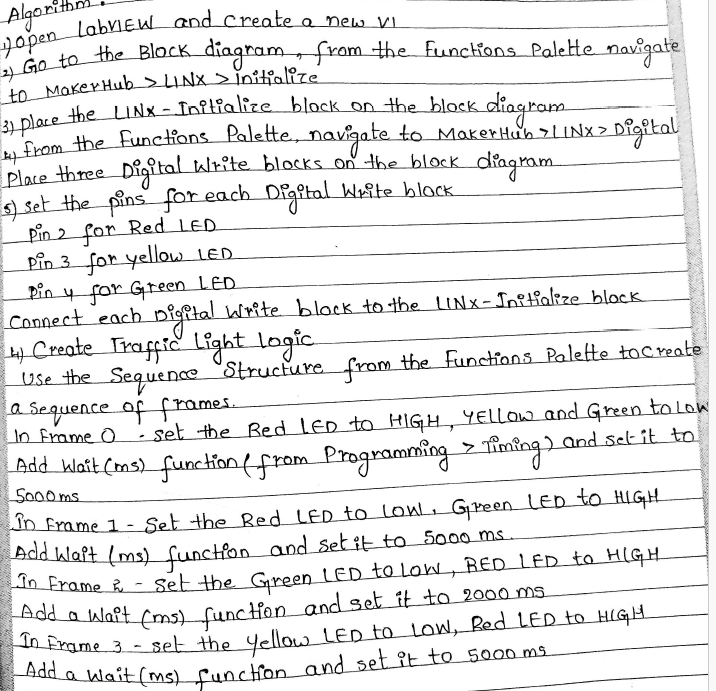
****

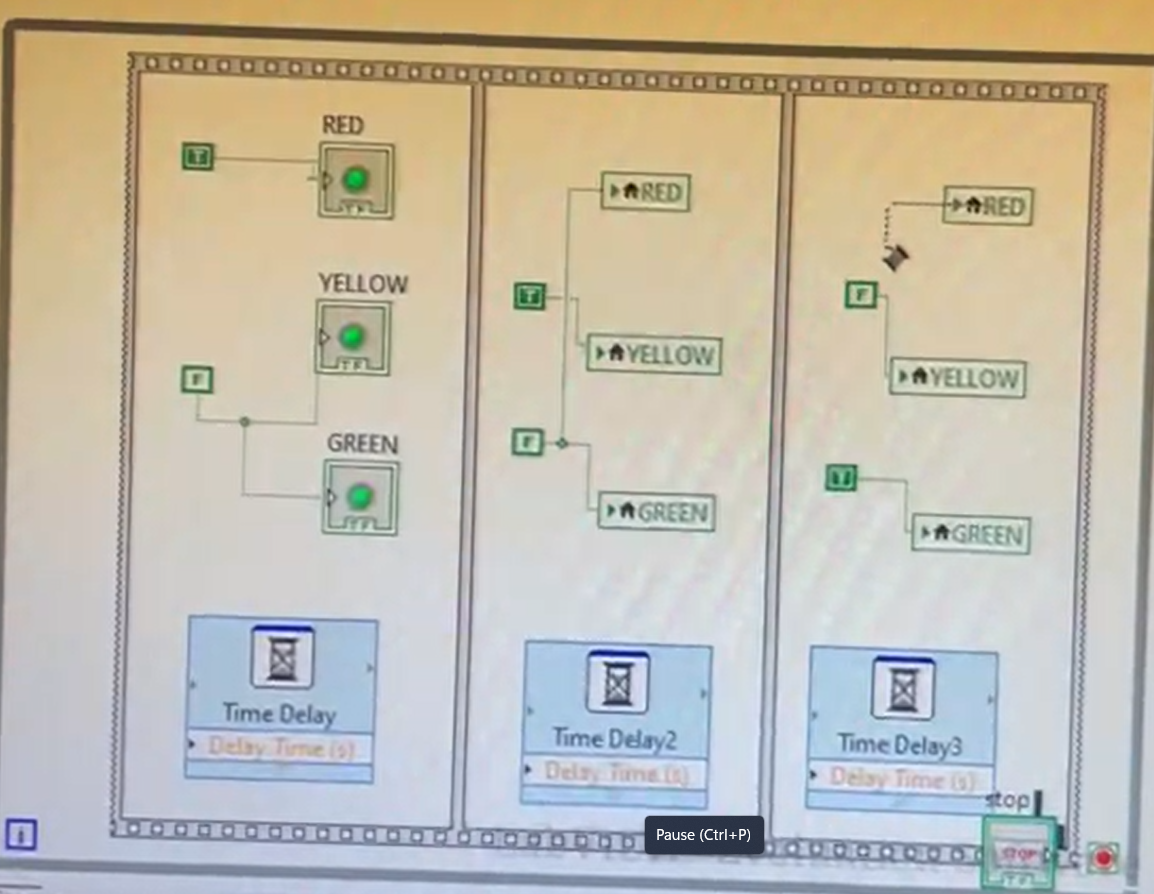
**8-on/off**

****

****

**9-traffic**

****

****