

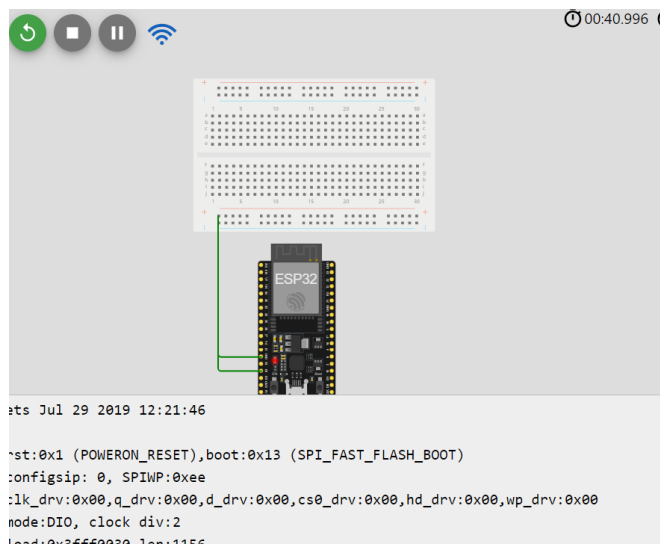
1. To write a program to sense the available networks using Arduino.

```
#include <WiFi.h>

void setup() {
  Serial.begin(115200); // Initialize Serial communication at 115200 baud rate
  delay(1000); // Give some time for the serial monitor to start
  Serial.println("Scanning for Wi-Fi networks...");
  // Start Wi-Fi in Station Mode
  WiFi.mode(WIFI_STA);
  WiFi.disconnect(); // Disconnect if already connected
  delay(100);
  // Start the scan
  int n = WiFi.scanNetworks();
  Serial.println("Scan complete!");
  if (n == 0) {
    Serial.println("No networks found.");
  } else {
    Serial.println(n + " networks found:");
    for (int i = 0; i < n; ++i) {
      // Print the SSID and signal strength for each found network
      Serial.print(i + 1);
      Serial.print(": ");
      Serial.print(WiFi.SSID(i));
      Serial.print(" (");
      Serial.print(WiFi.RSSI(i));
      Serial.print(" dBm)");
      Serial.println();
      delay(10); // Small delay to avoid flooding the Serial Monitor
    }
  }
}

void loop() {
  // Nothing to do here
}
```

Outout :-



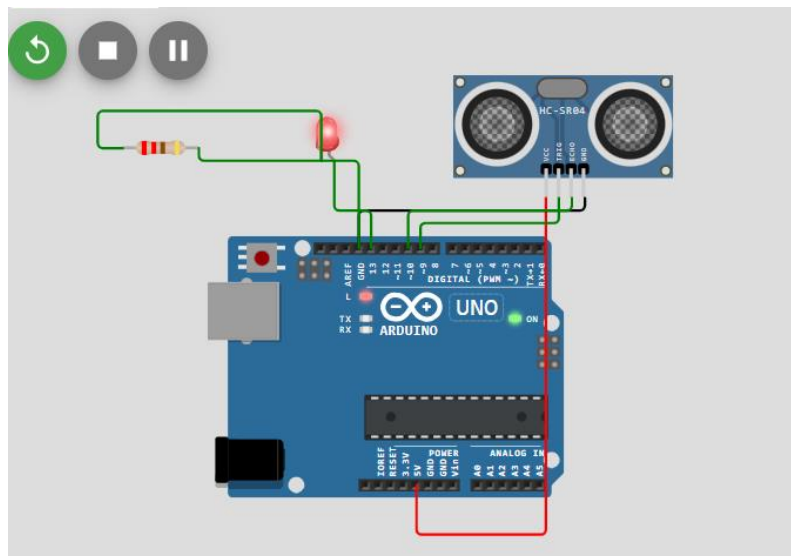
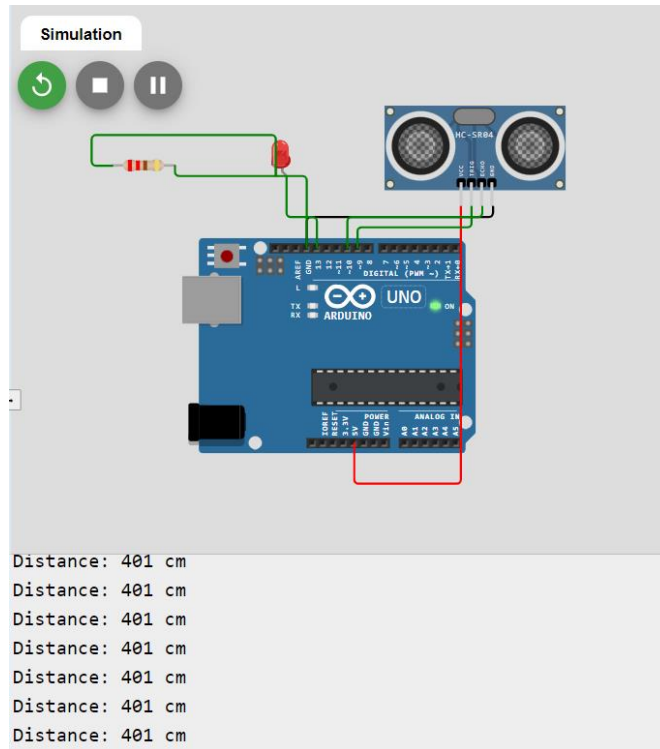
2. To write a program to measure the distance using ultrasonic sensor and make LED blink using Arduino.

```
const int trigPin = 9; // Trigger pin of the ultrasonic sensor
const int echoPin = 10; // Echo pin of the ultrasonic sensor
const int ledPin = 13; // LED pin
long duration;
int distance;
void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}
void loop() {
  // Clear the trigPin by setting it LOW
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Set the trigPin HIGH for 10 microseconds to send out the pulse
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  // Read the echoPin and calculate the time for the pulse to return
  duration = pulseIn(echoPin, HIGH);
  // Calculate the distance in cm (speed of sound is 343 m/s or 0.0343 cm/μs)
  distance = duration * 0.0343 / 2;
  // Print the distance to the serial monitor
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println(" cm");

  // Blink the LED if the distance is less than 10 cm
  if (distance < 410) {
    digitalWrite(ledPin, HIGH); // Turn on the LED
    delay(500); // Wait for half a second
    digitalWrite(ledPin, LOW); // Turn off the LED
    delay(500); // Wait for half a second
  }
  else {
    digitalWrite(ledPin, LOW); // Ensure LED is off when distance is above 10 cm
  }

  delay(200); // Add a small delay before repeating the loop
}
```

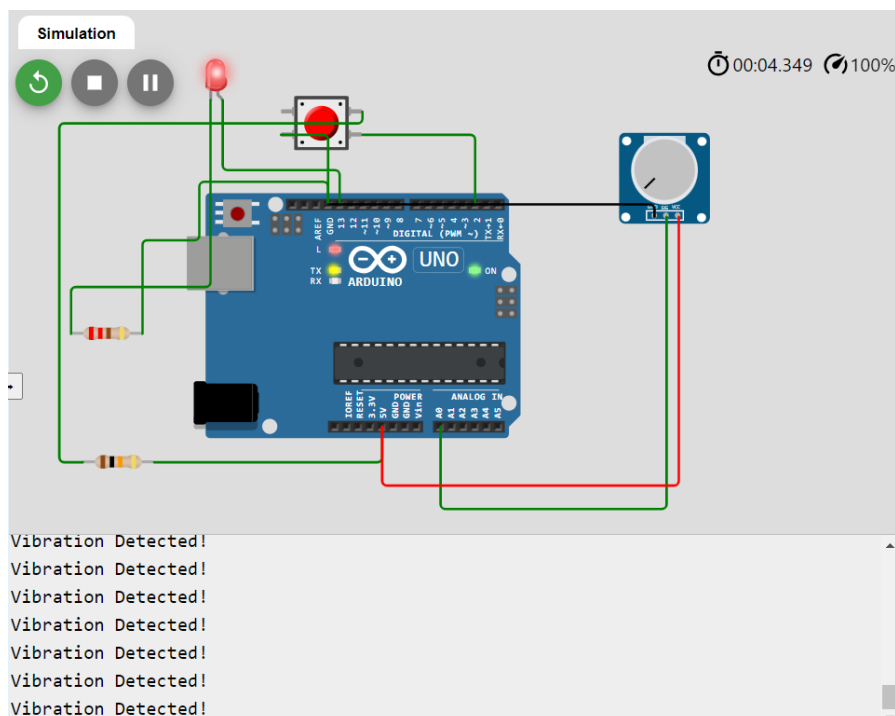
Output:-



3. To write a program to detects the vibration of an object with sensor using Arduino.

```
const int buttonPin = 2; // Pin connected to the push button
const int ledPin = 13;   // Pin connected to the LED
void setup() {
  pinMode(buttonPin, INPUT_PULLUP); // Set the button pin as input with internal
  pull-up resistor
  pinMode(ledPin, OUTPUT);          // Set the LED pin as output
  Serial.begin(9600);               // Start the serial communication
}
void loop() {
  int buttonState = digitalRead(buttonPin); // Read the state of the push button
  if (buttonState == LOW) { // If the button is pressed (acting like a vibration)
    digitalWrite(ledPin, HIGH); // Turn on the LED
    Serial.println("Vibration Detected!"); // Print message to serial monitor
  } else {
    digitalWrite(ledPin, LOW); // Turn off the LED
  }
  delay(100); // Small delay for stability
}
```

Output:-



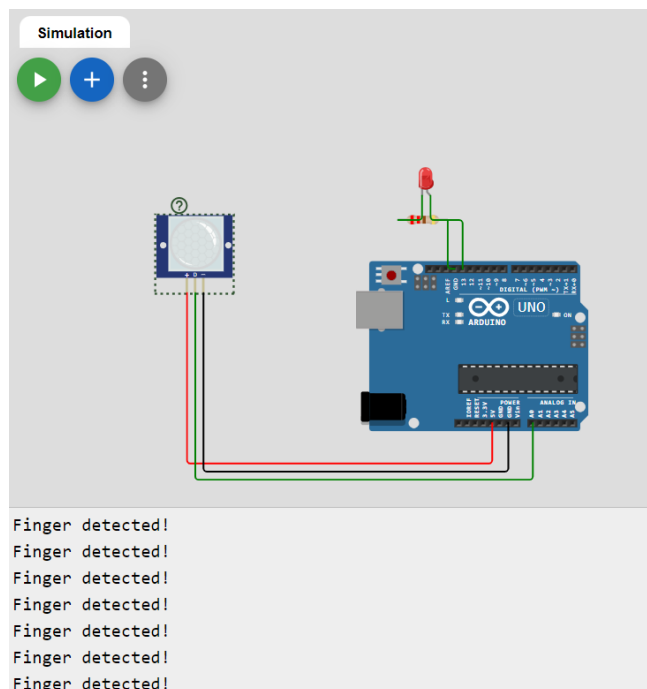
4. To write a program to sense a finger when it is placed on the board Arduino.

```
int sensorPin = A0; // IR sensor output connected to analog pin A0

void setup() {
  Serial.begin(115200);
}

void loop() {
  int sensorValue = analogRead(sensorPin); // Read the value from the sensor
  if (sensorValue < 100) { // Adjust threshold based on your setup
    Serial.println("Finger detected!");
  } else {
    Serial.println("No finger detected.");
  }
  delay(500); // Wait for half a second
}
```

Output :-



5. To write a program to connect with the available Wi-Fi using Arduino.

```
#include <WiFi.h> // Use <ESP8266WiFi.h> for ESP8266

const char* ssid = "your_SSID";    // Replace with your Wi-Fi SSID
const char* password = "your_PASSWORD"; // Replace with your Wi-Fi password
const int ledPin = 2; // Pin connected to the LED
const int timeout = 30; // Time out after 30 seconds

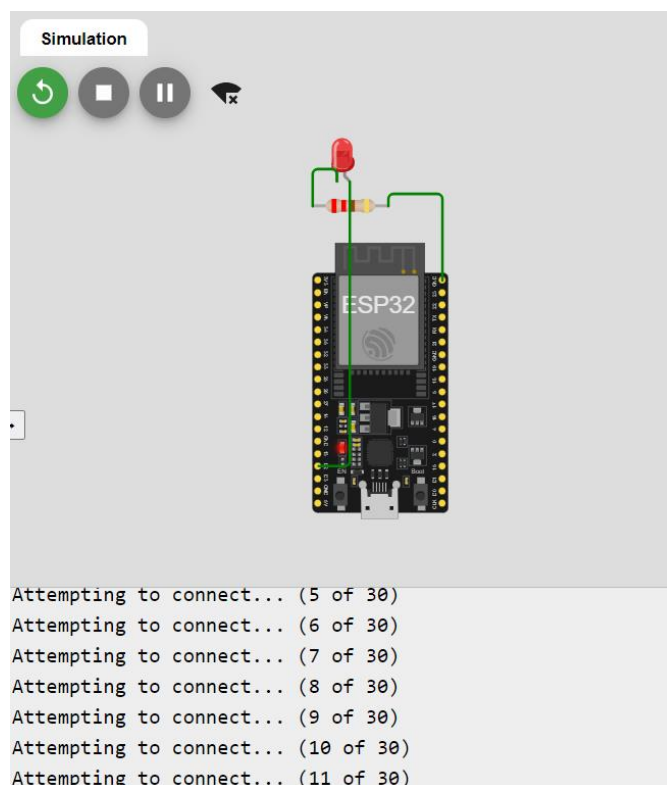
void setup() {
  Serial.begin(115200);
  pinMode(ledPin, OUTPUT); // Set LED pin as output
  delay(10);
  // Connecting to Wi-Fi
  Serial.println("Connecting to Wi-Fi...");
  while (true) { // Infinite loop for continuous connection attempts
    WiFi.begin(ssid, password);
    int attempts = 0;
    while (WiFi.status() != WL_CONNECTED && attempts < timeout) {
      delay(1000); // Wait for 1 second
      Serial.print("Attempting to connect... (");
      Serial.print(attempts + 1);
      Serial.print(" of ");
      Serial.print(timeout);
      Serial.println(")");
      attempts++;
    }
    // Check if connected
    if (WiFi.status() == WL_CONNECTED) {
      Serial.println("Connected to Wi-Fi!");
      Serial.print("IP address: ");
      Serial.println(WiFi.localIP()); // Print the IP address
```

```

digitalWrite(ledPin, HIGH); // Turn on the LED when connected
break; // Exit the loop if connected
} else {
    Serial.println("Failed to connect to Wi-Fi. Retrying...");
    // Blink LED to indicate failure
    for (int i = 0; i < 5; i++) {
        digitalWrite(ledPin, HIGH); // Turn on LED
        delay(500);
        digitalWrite(ledPin, LOW); // Turn off LED
        delay(500);
    }
    delay(5000); // Wait before retrying (5 seconds)
}
}
}
}
void loop() {
    // Optional: Additional code can be placed here
}

```

Output :-



6. To write a program to get temperature notification using Arduino.

```
// Define pin numbers for TM1637

#define CLK 2 // TM1637 CLK pin connected to Arduino pin 2
#define DIO 3 // TM1637 DIO pin connected to Arduino pin 3

// Define the temperature sensor pin (LM35)
const int tempPin = A0; // LM35 connected to A0

// Byte patterns for each digit on the 7-segment display
const byte digitPatterns[] = {
    0x3F, // 0
    0x06, // 1
    0x5B, // 2
    0x4F, // 3
    0x66, // 4
    0x6D, // 5
    0x7D, // 6
    0x07, // 7
    0x7F, // 8
    0x6F // 9
};

void setup() {
    pinMode(CLK, OUTPUT);
    pinMode(DIO, OUTPUT);
    Serial.begin(9600);
}

void loop() {
    int analogValue = analogRead(tempPin); // Read the analog value from LM35
    float voltage = analogValue * (5.0 / 1023.0); // Convert to voltage
    float temperatureC = voltage * 100.0; // Convert voltage to Celsius
    int displayTemp = (int)temperatureC;

    // Display the temperature on the TM1637 display
```



```

displayDigit(displayTemp / 10, 0); // Display tens digit
displayDigit(displayTemp % 10, 1); // Display ones digit
// Send temperature to Serial Monitor
Serial.print("Temperature: ");
Serial.print(temperatureC);
Serial.println("°C");
delay(1000); // Update every second
}

// Function to display a digit on TM1637
void displayDigit(int digit, int position) {
    startTransmission();
    writeByte(0x40); // Set automatic address increment mode
    stopTransmission();
    startTransmission();
    writeByte(0xC0 | position); // Set display address (0xC0 for 1st digit, 0xC1 for 2nd digit,
    etc.)
    writeByte(digitPatterns[digit]); // Send digit pattern
    stopTransmission();
    startTransmission();
    writeByte(0x8F); // Turn on display, max brightness
    stopTransmission();
}

void startTransmission() {
    digitalWrite(CLK, HIGH);
    digitalWrite(DIO, HIGH);
    digitalWrite(DIO, LOW);
    digitalWrite(CLK, LOW);
}

void stopTransmission() {
    digitalWrite(CLK, LOW);
    digitalWrite(DIO, LOW);
}

```

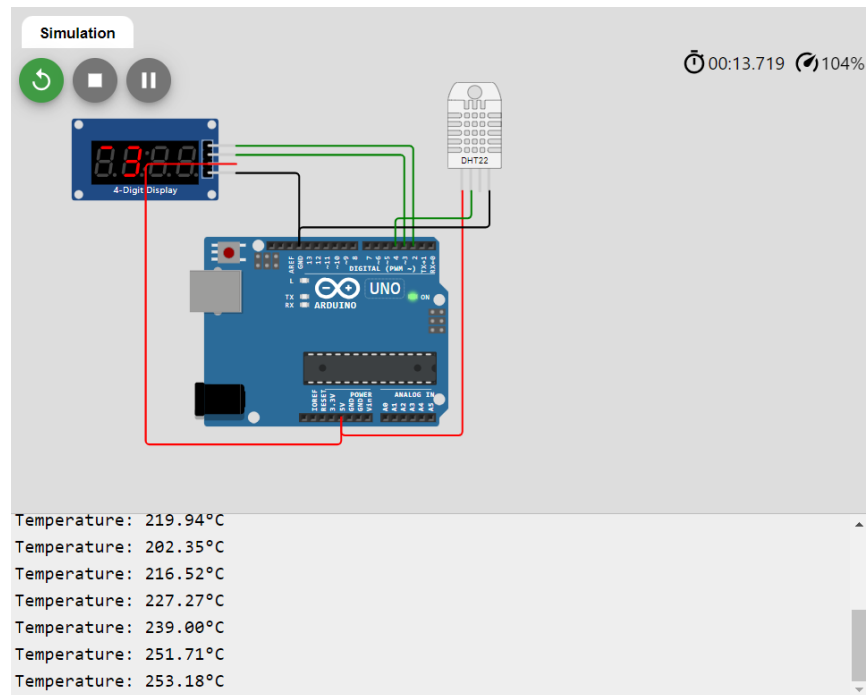
```

digitalWrite(CLK, HIGH);
digitalWrite(DIO, HIGH);
}

bool writeByte(byte b) {
  for (int i = 0; i < 8; i++) {
    digitalWrite(CLK, LOW);
    digitalWrite(DIO, (b & 0x01) ? HIGH : LOW);
    b >>= 1;
    digitalWrite(CLK, HIGH);
  }
  digitalWrite(CLK, LOW);
  pinMode(DIO, INPUT);
  digitalWrite(CLK, HIGH);
  bool ack = digitalRead(DIO) == 0;
  pinMode(DIO, OUTPUT);
  return ack;
}

```

Output :



```
const int ldrPin = A0; // LDR connected to A0

const int ledPin = 9; // LED connected to PWM pin 9

void setup() {
    pinMode(ledPin, OUTPUT); // Set LED pin as output
    Serial.begin(9600);    // Initialize serial communication for monitoring
}

void loop() {
    int ldrValue = analogRead(ldrPin); // Read the value from the LDR (0-1023)
    int brightness = map(ldrValue, 0, 1023, 0, 255); // Map LDR value to LED brightness (0-255)

    analogWrite(ledPin, brightness); // Set LED brightness based on LDR value
    // For debugging, print the LDR value and LED brightness
    Serial.print("LDR Value: ");
    Serial.print(ldrValue);
    Serial.print(" | LED Brightness: ");
    Serial.println(brightness);
    delay(100); // Delay for a short time
}
```

8. Start Raspberry Pi and try various Linux commands in command terminal window: ls, cd, touch, mv, rm, man, mkdir, rmdir, tar, gzip, cat, more, less, ps, sudo, cron, chown, chgrp, pingetc.

ls

```
Firefox Web Browser
onworks@onworks:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  Videos
onworks@onworks:~$
```

Cd

```
C:\Users\HRITIK\OneDrive\Desktop\hritik doc\practical> cd
C:\Users\HRITIK\OneDrive\Desktop\hritik doc\practical
```

PS

```
C:\Users\HRITIK\OneDrive\Desktop\hritik doc\practical\file>tasklist

Image Name                      PID Session Name        Session#    Mem Usage
=====
System Idle Process             0 Services             0           8 K
System                          4 Services             0        15,584 K
Registry                       104 Services            0        27,644 K
smss.exe                       552 Services            0         464 K
csrss.exe                      864 Services            0         2,872 K
wininit.exe                    952 Services            0         3,308 K
services.exe                   700 Services            0         6,600 K
lsass.exe                      868 Services            0        23,680 K
svchost.exe                   1136 Services            0        33,612 K
fontdrvhost.exe               1144 Services            0          888 K
WUDFHost.exe                  1216 Services            0         1,892 K
svchost.exe                   1288 Services            0        21,976 K
svchost.exe                   1356 Services            0         7,316 K
svchost.exe                   1436 Services            0         4,252 K
svchost.exe                   1452 Services            0         7,944 K
svchost.exe                   1492 Services            0         7,572 K
svchost.exe                   1520 Services            0         5,212 K
svchost.exe                   1528 Services            0         5,624 K
svchost.exe                   1748 Services            0         3,000 K
IntelCpHDCPSvc.exe           1756 Services            0         1,548 K
svchost.exe                   1816 Services            0         4,988 K
svchost.exe                   1812 Services            0         4,364 K
svchost.exe                   1828 Services            0        14,776 K
svchost.exe                   1940 Services            0         4,864 K
svchost.exe                   1952 Services            0        10,352 K
svchost.exe                   1624 Services            0         5,132 K
svchost.exe                   2052 Services            0         4,180 K
svchost.exe                   2096 Services            0         3,000 K
svchost.exe                   2160 Services            0         9,860 K
svchost.exe                   2168 Services            0         6,436 K
svchost.exe                   2360 Services            0        10,908 K
igfxCUIServiceN.exe          2368 Services            0         6,280 K
svchost.exe                   2468 Services            0         4,476 K
svchost.exe                   2484 Services            0         6,444 K
svchost.exe                   2668 Services            0        15,172 K
svchost.exe                   2744 Services            0         7,636 K
```

```
C:\Users\HRITIK\OneDrive\Desktop\hritik doc\practical\file>takeown /f Abc.txt
t

SUCCESS: The file (or folder): "C:\Users\HRITIK\OneDrive\Desktop\hritik doc\practical\file\Abc.txt" now owned by user "LAPTOP-N4TF9TOD\HRITIK".
```

ping

```
C:\Users\HRITIK\OneDrive\Desktop\hritik doc\practical\file>ping google.com

Pinging google.com [142.250.193.206] with 32 bytes of data:
Reply from 142.250.193.206: bytes=32 time=74ms TTL=109
Reply from 142.250.193.206: bytes=32 time=84ms TTL=109
Reply from 142.250.193.206: bytes=32 time=78ms TTL=109
Reply from 142.250.193.206: bytes=32 time=90ms TTL=109

Ping statistics for 142.250.193.206:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 74ms, Maximum = 90ms, Average = 81ms
```

```
PS C:\Users\Omkar\Desktop\new folder (2)> mkdir test_folder

Directory: C:\Users\Omkar\Desktop\new folder (2)

Mode                LastWriteTime         Length Name
----                -
d-----          10-11-2024   21:58             test_folder

PS C:\Users\Omkar\Desktop\new folder (2)> cd test_folder
PS C:\Users\Omkar\Desktop\new folder (2)\test_folder> echo "Hello Raspberry Pi" > example.txt
PS C:\Users\Omkar\Desktop\new folder (2)\test_folder> cat example.txt
Hello Raspberry Pi
PS C:\Users\Omkar\Desktop\new folder (2)\test_folder> cd ..
PS C:\Users\Omkar\Desktop\new folder (2)> rm -r test_folder
PS C:\Users\Omkar\Desktop\new folder (2)>
```

9. Run some python programs on Pi like:

a) Read your name and print Hello message with name

```
name = input("Enter your name: ")
print(f'Hello, {name}!')
```

Output :-

```
Enter your name: hritik
Hello, hritik!
Code cell output actions
```

b) Read two numbers and print their sum, difference, product and division.

```
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))
# Calculating sum, difference, product, and division
sum_result = num1 + num2
difference = num1 - num2
product = num1 * num2
division = num1 / num2 if num2 != 0 else "Undefined (division by zero)"
# Printing the results
print(f'Sum: {sum_result}')
print(f'Difference: {difference}')
print(f'Product: {product}')
print(f'Division: {division}')
```

Output :-

```
Enter the first number: 1
Enter the second number: 2
Sum: 3.0
Difference: -1.0
Product: 2.0
Division: 0.5
```

c) Word and character count of a given string.

c) Word and character count of a given string

```
text = input("Enter a string: ")
# Word count and character count
word_count = len(text.split()) # Splitting text by spaces to count words
char_count = len(text) # Counting all characters including spaces
print(f'Word Count: {word_count}')
print(f'Character Count: {char_count}')
```

Output :-

```
Enter a string: Hritik Thakar  
Word Count: 2  
Character Count: 13
```

d)Area of a given shape (rectangle, triangle and circle) reading shape and appropriate values from standard input.

```
import math  
  
# d) Area of a given shape (rectangle, triangle, and circle)  
  
def area_of_rectangle():  
    length = float(input("Enter the length of the rectangle: "))  
    width = float(input("Enter the width of the rectangle: "))  
    return length * width  
  
def area_of_triangle():  
    base = float(input("Enter the base of the triangle: "))  
    height = float(input("Enter the height of the triangle: "))  
    return 0.5 * base * height  
  
def area_of_circle():  
    radius = float(input("Enter the radius of the circle: "))  
    return math.pi * radius * radius  
  
# Asking for the shape  
shape = input("Enter the shape (rectangle, triangle, or circle): ").lower()  
  
# Calculating the area based on the shape  
if shape == "rectangle":  
    area = area_of_rectangle()  
    print(f"The area of the rectangle is: {area}")  
elif shape == "triangle":  
    area = area_of_triangle()  
    print(f"The area of the triangle is: {area}")  
elif shape == "circle":  
    area = area_of_circle()
```

```
    print(f"The area of the circle is: {area}")
else:
    print("Invalid shape entered.")
```

Output :-

```
Enter the shape (rectangle, triangle, or circle): rectangle
Enter the length of the rectangle: 12
Enter the width of the rectangle: 24
The area of the rectangle is: 288.0
```

10. Run some python programs on Pi like:

a) Print a name 'n' times, where name and n are read from standard input, using for and while loops.

```
name = input("Enter your name: ")
n = int(input("How many times do you want to print your name? "))
# Using a for loop
for i in range(n):
    print(name)
```

```
name = input("Enter your name: ")
n = int(input("How many times do you want to print your name? "))
# Using a while loop
count = 0
while count < n:
    print(name)
    count += 1
```

Output :-

```
Enter your name: Hritik
How many times do you want to print your name? 3
Hritik
Hritik
Hritik
Enter your name: Hritik
How many times do you want to print your name? 3
Hritik
Hritik
Hritik
```


b) Handle Divided by Zero Exception.

b) Handle divided by zero exception

try:

```
num1 = float(input("Enter the numerator: "))
num2 = float(input("Enter the denominator: "))
result = num1 / num2
print(f"The result is: {result}")
```

except ZeroDivisionError:

```
print("Error: Division by zero is not allowed.")
```

Output :-

```
Enter the numerator: 2
Enter the denominator: 1
The result is: 2.0
```

c) Print current time for 10 times with an interval of 10seconds.

import time

from datetime import datetime

c) Print current time for 10 times with an interval of 10 seconds

for i in range(10):

```
    current_time = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
    print(f"Current time: {current_time}")
    time.sleep(10)
```

Output :-

```
* Current time: 2024-10-22 15:26:37
Current time: 2024-10-22 15:26:47
Current time: 2024-10-22 15:26:57
Current time: 2024-10-22 15:27:07
Current time: 2024-10-22 15:27:17
```

d) Read a fileline byline and print the word count of each line

Create a sample file to read

with open("sample.txt", "w") as file:

```
    file.write("Hello world\n")
    file.write("This is a test file\n")
    file.write("Python is fun to learn\n")
```

Read a file line by line and print word count of each line

```
with open("sample.txt", "r") as file:
    for i, line in enumerate(file, start=1):
        word_count = len(line.split())
        print(f'Line {i}: {word_count} words')
```

Output :-

```
Line 1: 2 words
Line 2: 5 words
Line 3: 5 words
```

11. Run some python programs on Pi like

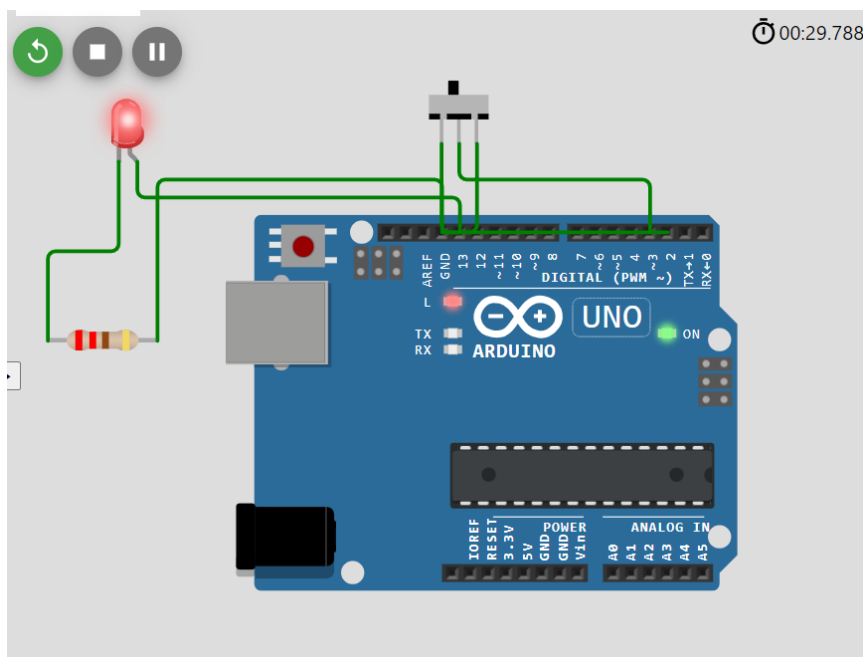
a) Light an LED through Python program

```
int ledPin = 13; // LED connected to digital pin 13

void setup() {
    pinMode(ledPin, OUTPUT); // Set pin 13 as output
}

void loop() {
    digitalWrite(ledPin, HIGH); // Turn the LED on
    delay(2000);                // Wait for a second
    digitalWrite(ledPin, LOW);  // Turn the LED off
    delay(5000);                // Wait for a second
```

Output :-



b) Get input from two switches and switch on corresponding LEDs

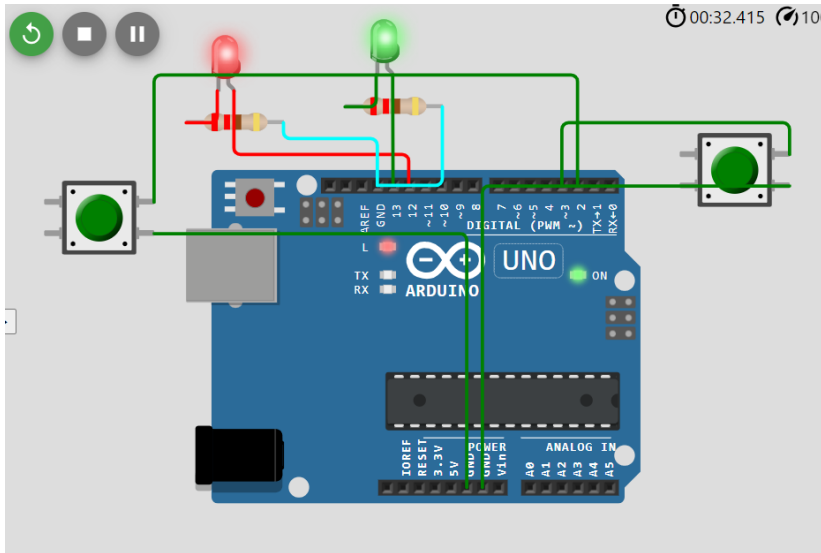
```
int switch1Pin = 2; // Switch 1 connected to digital pin 2
int switch2Pin = 3; // Switch 2 connected to digital pin 3
int led1Pin = 12; // LED 1 connected to digital pin 12
int led2Pin = 13; // LED 2 connected to digital pin 13

void setup() {
  pinMode(switch1Pin, INPUT); // Set switch 1 pin as input
  pinMode(switch2Pin, INPUT); // Set switch 2 pin as input
  pinMode(led1Pin, OUTPUT); // Set LED 1 pin as output
  pinMode(led2Pin, OUTPUT); // Set LED 2 pin as output
}

void loop() {
  if (digitalRead(switch1Pin) == HIGH) {
    digitalWrite(led1Pin, HIGH); // Turn on LED 1 if switch 1 is pressed
  } else {
    digitalWrite(led1Pin, LOW); // Turn off LED 1 otherwise
  }

  if (digitalRead(switch2Pin) == HIGH) {
    digitalWrite(led2Pin, HIGH); // Turn on LED 2 if switch 2 is pressed
  } else {
    digitalWrite(led2Pin, LOW); // Turn off LED 2 otherwise
  }
}
```

Output :-



c) Flash an LED at a given on time and off time cycle, where the two times are taken from a file

```
int ledPin = 13; // LED connected to digital pin 13

int onTime = 100; // On time in milliseconds

int offTime = 1000; // Off time in milliseconds

void setup() {
  pinMode(ledPin, OUTPUT); // Set LED pin as output
}

void loop() {
  digitalWrite(ledPin, HIGH); // Turn the LED on
  delay(onTime);              // Wait for onTime duration
  digitalWrite(ledPin, LOW);  // Turn the LED off
  delay(offTime);
}
```

Output :-

