

## Listing Program Arduino IDE

```
#include <Adafruit_Fingerprint.h>
#include <Keypad.h>
#include <SoftwareSerial.h>
#include <LiquidCrystal.h>
#include <Password.h>

SoftwareSerial mySerial(2, 3);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);

byte sensorsentuh= 13;
int nilaisensor;
byte irf= 12;
int idFinger;
byte currentLength = 0;
#define irf_ON 1
#define irf_OFF 0
#define irf_1 12

boolean currentState = HIGH;
boolean lastState = HIGH;
boolean irfState = HIGH;
Password password = Password("2016");//Password bisa di ubah ubah

LiquidCrystal lcd(A0, A1, A2, A3, A4, A5);// pin lcd yang di gunakan
```

```
const byte ROWS= 4; //baris pada keypad
const byte COLS= 4; //Kolom pada keypad
```

```
//inisialisasi led dan irf sebagai output bila password betul maka akan menyala irf,
//dan bila password salah akan menyala led merah
```

```
/*keymap mendefinisikan tombol ditekan sesuai
dengan baris dan kolom seperti muncul pada keypad*/
```

```
char keys[ROWS][COLS] =
{
{'1', '2', '3', 'A'},
{'4', '5', '6', 'B'},
{'7', '8', '9', 'C'},
{'*', '0', '#', 'D'}
};
```

```
byte rowPins[ROWS]= {11,10,9,8};
```

```
byte colPins[COLS]= {7,6,5,4};
```

```
Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
```

```
void setup(){
```

```
  Serial.begin(9600);
```

```
finger.begin(57600);  
keypad.addEventListener(keypadEvent);
```

```
lcd.begin(16, 2);  
lcd.setCursor(0, 0);  
lcd.print("DoorlockOtomatis");  
lcd.setCursor(5, 1);  
lcd.print("T & A");  
delay(4000);  
lcd.clear();  
lcd.setCursor(1, 0);  
lcd.print("Selamat Datang");  
lcd.setCursor(3, 1);  
lcd.print("Di Tekkom");  
delay(4000);  
lcd.clear();
```

```
pinMode(sensorsentuh, INPUT);  
pinMode(irf, OUTPUT);
```

```
if (finger.verifyPassword()) {  
  Serial.println("FingerPrint Sensor Ditemukan!");  
} else {  
  Serial.println("FingerPrint Sensor Tidak Ditemukan! :(");  
  while (1) { delay(1); }  
}
```

```
}
```

```
void loop(){  
  FINGERPRINT();  
  nilaisensor= digitalRead(sensorsentuh);  
  Serial.println(nilaisensor);
```

```
  
  lcd.setCursor(1, 0);  
  lcd.print("Masukan Akses");  
  lcd.setCursor(0,1);  
  lcd.print(" ");  
  keypad.getKey();;
```

```
  
  if(idFinger >= 0){  
    lcd.clear();  
    lcd.setCursor(0, 0);  
    lcd.print("Akses Sidik Jari");  
    lcd.setCursor(4, 1);  
    lcd.print("Diterima");  
    digitalWrite(irf, HIGH);  
    delay(5000);  
    lcd.clear();  
    lcd.setCursor(1, 0);  
    lcd.print("Masukan Akses");  
    lcd.setCursor(0,1);  
    lcd.print(" ");  
    keypad.getKey();;
```

```
    digitalWrite(irf, LOW);
}
if(nilaisensor == 1){
    lcd.clear();
    lcd.setCursor(2, 0);
    lcd.print("Terbuka dari");
    lcd.setCursor(4, 1);
    lcd.print("Dalam");
    digitalWrite(irf, HIGH);
    delay(5000);
    lcd.clear();
    lcd.setCursor(1, 0);
    lcd.print("Masukan Akses");
    lcd.setCursor(0,1);
    lcd.print(" ");
    keypad.getKey();;
    digitalWrite(irf, LOW);
}

}
```

```
//-----KEYPAD-----//
void checkPassword(){
    if(password.evaluate()){
        digitalWrite(irf_1, irf_ON);
        lcd.clear();
        lcd.setCursor(3,0);
        lcd.print("Akses Sandi");
    }
}
```

```

    lcd.setCursor(4,1);
    lcd.print("Diterima");
    Serial.println("Berhasil Dari keypad");
    delay(5000);//Lama waktu irf
    password.reset();
    digitalWrite(irf_1, irf_OFF);
}
else {

    lcd.clear();
    lcd.setCursor(1, 0);
    lcd.print("Akses Ditolak");
    lcd.setCursor(0
, 1);
    lcd.print("Salah coba lagi");
    delay(2000);//lama led on
    password.reset();

}
}

void keypadEvent(KeypadEvent eKey){

    switch (keypad.getState()){
        case PRESSED:
            lcd.setCursor(0,1);
            lcd.print(eKey);
            Serial.println(eKey);

```

```

switch (eKey){
    case '*': checkPassword();
    lcd.clear(); currentLength=0;break;
    case '#': password.reset();
    lcd.clear();currentLength=0; break;
    default:// password.append(eKey);
        password << eKey;
        currentLength++;

        //Print some feedback.
        lcd.setCursor(0,1);
        lcd.print(" ");
        for (byte i=0; i<currentLength; i++){
            lcd.print('*');
        }

    }
}
}

//-----FINGERPRINT-----
---//

void FINGERPRINT(){
    idFinger= getFingerprintIDez();
    delay(50);
}

//-----PROSES FINGERPRINT-----
-----//

uint8_t getFingerprintID() {

```

```
uint8_t p = finger.getImage();
switch (p) {
  case FINGERPRINT_OK:
    Serial.println("Image taken");
    break;
  case FINGERPRINT_NOFINGER:
    Serial.println("No finger detected");
    return p;
  case FINGERPRINT_PACKETRECEIVEERR:
    Serial.println("Communication error");
    return p;
  case FINGERPRINT_IMAGEFAIL:
    Serial.println("Imaging error");
    return p;
  default:
    Serial.println("Unknown error");
    return p;
}
```

```
// OK success!
```

```
p = finger.image2Tz();
switch (p) {
  case FINGERPRINT_OK:
    Serial.println("Image converted");
    break;
  case FINGERPRINT_IMAGEMESS:
    Serial.println("Image too messy");
```



```
    return p;
case FINGERPRINT_PACKETRECIEVEERR:
    Serial.println("Communication error");
    return p;
case FINGERPRINT_FEATUREFAIL:
    Serial.println("Could not find fingerprint features");
    return p;
case FINGERPRINT_INVALIDIMAGE:
    Serial.println("Could not find fingerprint features");
    return p;
default:
    Serial.println("Unknown error");
    return p;
}

// OK converted!
p = finger.fingerFastSearch();
if (p == FINGERPRINT_OK) {
    Serial.println("Found a print match!");
} else if (p == FINGERPRINT_PACKETRECIEVEERR) {
    Serial.println("Communication error");
    return p;
} else if (p == FINGERPRINT_NOTFOUND) {
    Serial.println("Did not find a match");
    return p;
} else {
    Serial.println("Unknown error");
    return p;
}
```

```

}

// found a match!
Serial.print("Found ID #"); Serial.print(finger.fingerID);
Serial.print(" with confidence of "); Serial.println(finger.confidence);

return finger.fingerID;
}

// returns -1 if failed, otherwise returns ID #
int getFingerprintIDez() {
  uint8_t p = finger.getImage();
  if (p != FINGERPRINT_OK) return -1;

  p = finger.image2Tz();
  if (p != FINGERPRINT_OK) return -1;

  p = finger.fingerFastSearch();
  if (p != FINGERPRINT_OK) return -1;

  // found a match!
  Serial.print("Found ID #"); Serial.print(finger.fingerID);
  Serial.print(" with confidence of "); Serial.println(finger.confidence);
  return finger.fingerID;
}

//-----END FINGERPRINT-----//

```