LAMPIRAN-LAMPIRAN

***Coding* program “Prototipe Kandang Ayam *Broiler* Dengan Kontrol Suhu Dan Kelembaban Berbasis *Arduino Mega 2560 Pro* Serta *SMS Gateway*” dan Tabel data suhu, kelembaban dan status output.**

NAMA	:	HERU PURNOMO
NIM	:	141220000029
PRODI	:	TEKNIK ELEKTRO
FAKULTAS	:	SAINS DAN TEKNOLOGI UNIVERSITAS ISLAM NAHDLATUL ULAMA JEPARA

Lampiran 1: *Coding* program

```
//****************************************************
#include "U8glib.h"
#include <EEPROM.h>
// Date and time functions using a DS1307 RTC connected via I2C
and Wire lib
#include <Wire.h>
#include "RTClib.h"
#include <Sim800L.h>
#include <SoftwareSerial.h>
#include <SPI.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>
#include <SD.h>
// Include the Bounce2 library found here :
// https://github.com/thomasfredericks/Bounce2
#include <Bounce2.h>

#define BUTTON_PIN 22
// Instantiate a Bounce object
Bounce debouncer = Bounce();
byte buttonState;
uint32_t buttonPressTimeStamp;

const int buttonUP = 24;    // the number of the pushbutton pin
const int buttonDN = 25;    // the number of the pushbutton pin
int buttonStateUP=LOW;   // the current reading from the input pin
int buttonStateDN=LOW;
int lastButtonStateUP = LOW;   // the previous reading from the
input pin
int lastButtonStateDN = LOW;
// the following variables are unsigned long's because the time,
measured in miliseconds,
// will quickly become a bigger number than can be stored in an
int.
unsigned long lastDebounceTime = 0;  // the last time the output
pin was toggled
unsigned long debounceDelay = 50;    // the debounce time;
increase if the output flickers

//Create software serial object to communicate with SIM800L
SoftwareSerial mySerial(13, 12); //SIM800L Tx & Rx is connected to
Arduino #13 & #12

  U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NONE);  // I2C
  const int chipSelect = 53;  // change this to match your SD
shield or module;
  // Create a file to store the data
  File myFile;

  Adafruit_BME280 bme;    // I2C BME280
//  unsigned long delayTime;
  unsigned long
smspreviousMillis=0,jampreviousMillis=0,logpreviousMillis=0;
```

```
  unsigned long jaminterval = 60UL*60UL*1000UL;    //millis 1 jam
  unsigned long loginterval = 5UL*1000UL;    //millis logging 5
detik
  unsigned long smsinterval = 15UL*60UL*1000UL;    //millis sms 15
menit
  uint32_t bmepreviousMillis=0,waktupreviousMillis=0;
  uint32_t bmeinterval = 1000*3;      //millis baca sensor BME
  uint32_t waktuinterval = 1000*1;    //millis display OLED
  float humidity=0,temperature=0,toleransi=0;
  String thisTemp = "",thisHumidity = "",thisAbntext="";

  RTC_DS1307 RTC;
  String thisMonth = "";
  String thisTime = "";
  String thisTime1 = "";
  String thisDay = "";
  String thisFile = "";
  String output,str = "";

  byte hari,jam,jalan;
  byte cd,cu;            //batas suhu bawah dan atas
  byte hd=60,hu=70;      //batas kelembaban bawah 60 dan atas 70

  const int buzzer=23;      //buzzer
  const int pelembab=32;    //pelembab
  const int pengering=33;   //pengering
  const int pemanas=34;     //pemanas
  const int pendingin=35;   //pendingin

void setup(void) {
  Serial.begin(9600);
  while (!Serial) ; // wait for serial
  //Begin serial communication with Arduino and SIM800L
  mySerial.begin(9600);

  Wire.begin();
  RTC.begin();
  delay(2000);
  if (! RTC.isrunning()) {
    Serial.println("RTC is NOT running!");
    // following line sets the RTC to the date & time this sketch
was compiled
    //RTC.adjust(DateTime(__DATE__, __TIME__));
    delay(1000);
  }

  // setup for the SD card
  Serial.print("Initializing SD card...");
  if(!SD.begin(chipSelect)) {
    Serial.println("initialization failed!");
 }
  Serial.println("initialization done.");

  // Setup the button
  pinMode(BUTTON_PIN,INPUT);
  // Activate internal pull-up
  digitalWrite(BUTTON_PIN,HIGH);
  // After setting up the button, setup debouncer
  debouncer.attach(BUTTON_PIN);
  debouncer.interval(5);

  //configure pins as an input and enable the internal pull-up
resistor
  pinMode(buttonUP, INPUT_PULLUP);
  pinMode(buttonDN, INPUT_PULLUP);
  buttonStateUP=HIGH;
  buttonStateDN=HIGH;
  pinMode(buzzer, OUTPUT);
  digitalWrite(buzzer, LOW);
  pinMode(pemanas, OUTPUT);
  digitalWrite(pemanas, LOW);
  pinMode(pendingin, OUTPUT);
  digitalWrite(pendingin, LOW);
  pinMode(pelembab, OUTPUT);
  digitalWrite(pelembab, LOW);
  pinMode(pengering, OUTPUT);
  digitalWrite(pengering, LOW);

    unsigned status;
    // default settings
    status = bme.begin();
    // You can also pass in a Wire library object like &Wire2
    // status = bme.begin(0x76, &Wire2)
    status = bme.begin(0x76);
    if (!status) {
        Serial.println("Could not find a valid BME280 sensor,
check wiring, address, sensor ID!");
        Serial.print("SensorID was: 0x");
Serial.println(bme.sensorID(),16);
        Serial.print("        ID of 0xFF probably means a bad
address, a BMP 180 or BMP 085\n");
        Serial.print("        ID of 0x56-0x58 represents a BMP
280,\n");
        Serial.print("        ID of 0x60 represents a BME
280.\n");
        Serial.print("        ID of 0x61 represents a BME
680.\n");
        while (1) delay(10);
    }
    Serial.println("-- BME Default Test --");
    Serial.println();
}

void printValues() {
    Serial.print("Temperature = ");
    Serial.print(bme.readTemperature());
    Serial.println(" *C");
    Serial.print("Humidity = ");
    Serial.print(bme.readHumidity());
    Serial.println(" %");
    Serial.println();
}
```

```
void loop(void) {
  jalan=EEPROM.read(0);
  hari=EEPROM.read(1);
  jam=EEPROM.read(2);
  if (hari<3) {
      cd=29;
      cu=32;    //memasukkan data suhu atas 32 dan bawah 29
  }
  else if (hari>=3 and hari<6) {
      cd=27;
      cu=30;    //memasukkan data suhu atas 30 dan bawah 27
  }
  else if (hari>=6 and hari<9) {
      cd=25;
      cu=28;    //memasukkan data suhu atas 28 dan bawah 25
  }
  else if (hari>=9 and hari<12) {
      cd=25;
      cu=27;    //memasukkan data suhu atas 27 dan bawah 25
  }
  else if (hari>=12 and hari<15) {
      cd=25;
      cu=26;    //memasukkan data suhu atas 26 dan bawah 25
  }
  else {
      cd=24;
      cu=25;    //memasukkan data suhu atas 25 dan bawah 24
  }

//-------tombol start & stop------------
 // Update the debouncer and get the changed state
  boolean changed = debouncer.update();
  if ( changed ) {
    // Get the update value
    int value = debouncer.read();
    if ( value == HIGH) {
       buttonState = 0;
       //Serial.println("Button released (state 0)");
   } else {
       buttonState = 1;
       //Serial.println("Button pressed (state 1)");
       buttonPressTimeStamp = millis();
   }
  }
  if  ( buttonState == 1 ) {
    if ( millis() - buttonPressTimeStamp >= 3000 ) {
      buttonPressTimeStamp = millis();
      if (jalan>0) {
          henti_program();
      } else {
          jalan_program();
      }
    }
  }
```

```
    //***** RTC **********
  DateTime now = RTC.now();
  // ********************
  thisDay ="";
  thisFile="";
  if (now.day() < 10){ thisDay=thisDay + "0"; } // add leading
zero if required
  thisDay = thisDay+String(now.day(), DEC);
  //thisFile = thisDay;
  thisMonth="";
  if (now.month() < 10){thisMonth=thisMonth+ "0";} // add leading
zero if required
  thisMonth = thisMonth+String(now.month(), DEC);
  thisDay=thisMonth + "-" + thisDay;
  thisFile=thisMonth + thisFile;
  thisDay=String(now.year() , DEC) + "-" + thisDay;
  thisFile=String(now.year() , DEC) + thisFile;
  // ********************
  thisTime="";
  if (now.hour() < 10){thisTime=thisTime + "0";} // add leading
zero if required
  thisTime=thisTime+String(now.hour()) + ":";
  if (now.minute() < 10){thisTime=thisTime + "0";} // add leading
zero if required
  thisTime1=thisTime + String(now.minute()) ;
  thisTime=thisTime + String(now.minute()) + ":";
  if (now.second() < 10){ thisTime=thisTime + "0";} // add leading
zero if required
  thisTime=thisTime + String(now.second());

  //-------millis sensor BME--------
  unsigned long bmecurrentMillis = millis();
  if(bmecurrentMillis - bmepreviousMillis > bmeinterval)
  {
    bmepreviousMillis = bmecurrentMillis;

  temperature = bme.readTemperature();
  humidity = bme.readHumidity();
  Serial.println();
  Serial.println(thisDay+","+thisTime);
  Serial.println(temperature);
  Serial.println(humidity);

  if (jalan>0) {
  Serial.print(hari);
  Serial.print(" hari, ");
  Serial.print(jam);
  Serial.println(" jam");

  thisAbntext="";
  //thisAbntext="Kondisi ";
  //urutan log output = pemanas,pendingin,pelembab,pengering
        output=" ";
        output="hari ke=,"+String(hari)+",suhu
acuan=,"+String(cd)+","+String(cu)+",output=";
  //cek batas suhu bawah
  if(temperature < (cd + toleransi))  {
    thisAbntext=thisAbntext+"terlalu DINGIN ";
    Serial.println("pemanas ON");
    output=output+",1";
    digitalWrite(pemanas,HIGH); //nyalakan pemanas
    }
    else {
    Serial.println("pemanas OFF");
    output=output+",0";
    digitalWrite(pemanas,LOW);  //matikan pemanas
    }
  //cek batas suhu atas
  if(temperature > (cu - toleransi)) {
    thisAbntext=thisAbntext+"terlalu PANAS ";
    Serial.println("pendingin ON");
    output=output+",1";
    digitalWrite(pendingin,HIGH); //nyalakan pendingin
    }
    else {
    Serial.println("pendingin OFF");
    output=output+",0";
    digitalWrite(pendingin,LOW);  //matikan pendingin
    }
  //cek batas kelembaban bawah
  if(humidity < (hd + toleransi)) {
    thisAbntext=thisAbntext+"terlalu KERING ";
    Serial.println("pelembab ON");
    output=output+",1";
    digitalWrite(pelembab,HIGH); //nyalakan pelembab
    }
    else {
    Serial.println("pelembab OFF");
    output=output+",0";
    digitalWrite(pelembab,LOW);  //matikan pelembab
    }
  //cek batas kelembaban atas
  if(humidity > (hu - toleransi)) {
    thisAbntext=thisAbntext+"terlalu LEMBAB ";
    Serial.println("pengering ON");
    output=output+",1";
    digitalWrite(pengering,HIGH); //nyalakan pengering
    }
    else {
    Serial.println("pengering OFF");
    output=output+",0";
    digitalWrite(pengering,LOW);  //matikan pengering
    }

        Serial.println(output);
    }
  }

  if (jalan>0) {
  //---------millis counter jam & hari-------
```

```
  uint32_t jamcurrentMillis = millis();
  if(jamcurrentMillis - jampreviousMillis > jaminterval)
    {
      jampreviousMillis = jamcurrentMillis;
      hari_jam ();
    }
  //---------millis data logger sdcard--------
  uint32_t logcurrentMillis = millis();
  if(logcurrentMillis - logpreviousMillis > loginterval)
    {
      logpreviousMillis = logcurrentMillis;
      logger ();
    }
  //---------millis kirim sms--------
  unsigned long smscurrentMillis = millis();
  if(smscurrentMillis - smspreviousMillis > smsinterval)
    {
    smspreviousMillis = smscurrentMillis;
      if (digitalRead(pemanas)==HIGH or
digitalRead(pendingin)==HIGH or digitalRead(pengering)==HIGH or
digitalRead(pelembab)==HIGH)
      {
          thisAbntext="Peringatan!! Kondisi "+thisAbntext;
      } else
      {
          thisAbntext="Dalam kondisi NORMAL";
      }
    kirim_sms ();
    }

  // read the state of the switch into a local variable:
  int readingUP = digitalRead(buttonUP);
  int readingDN = digitalRead(buttonDN);
  // check to see if you just pressed the button
  // (i.e. the input went from LOW to HIGH), and you've waited
  // long enough since the last press to ignore any noise:
  // If the switch changed, due to noise or pressing:
  if ((readingUP != lastButtonStateUP) or (readingDN !=
lastButtonStateDN)){
    // reset the debouncing timer
    lastDebounceTime = millis();
  }
  if ((millis() - lastDebounceTime) > debounceDelay) {
    // whatever the reading is at, it's been there for longer
    // than the debounce delay, so take it as the actual current
state:
    // if the button state has changed:
    if (readingUP != buttonStateUP) {
      buttonStateUP = readingUP;
      // only toggle the LED if the new button state is HIGH
      if (buttonStateUP == HIGH) {
        hari=EEPROM.read(1);
        if (hari<50) {
          hari++;
          EEPROM.update(1,hari);
          digitalWrite(buzzer,HIGH);
          delay(50);
          digitalWrite(buzzer,LOW);
        }
      }
    }
    // if the button state has changed:
    if (readingDN != buttonStateDN) {
      buttonStateDN = readingDN;
      // only toggle the LED if the new button state is HIGH
      if (buttonStateDN == HIGH) {
        hari=EEPROM.read(1);
        if (hari>1) {
          hari--;
          EEPROM.update(1,hari);
          digitalWrite(buzzer,HIGH);
          delay(50);
          digitalWrite(buzzer,LOW);
        }
      }
    }
  }
  // save the reading.  Next time through the loop,
  // it'll be the lastButtonState:
  lastButtonStateUP = readingUP;
  lastButtonStateDN = readingDN;
 }

//---------millis tampilan jam di oled--------
  uint32_t waktucurrentMillis = millis();
  if(waktucurrentMillis - waktupreviousMillis > waktuinterval)
    {
      waktupreviousMillis = waktucurrentMillis;
      // picture loop
      u8g.firstPage();
      do {
        draw();
      } while( u8g.nextPage() );
    }

}

void hari_jam (void) {    //counter jam
  jalan=EEPROM.read(0);
if (jalan<1){
   }
else {
  hari=EEPROM.read(1);
  jam=EEPROM.read(2);
  jam++;
  if (jam>23){
    jam=0;
    hari++;
    }
  EEPROM.update(2,jam);
  EEPROM.update(1,hari);
  Serial.print(thisDay+","+thisTime+", counter per 1 jam");
  Serial.println();
  }
}

void logger (void){
  jalan=EEPROM.read(0);
  if (jalan>0){
  Serial.println();
  Serial.print(thisDay+","+thisTime);
  Serial.print(","+thisTemp+"'C");
  Serial.print(","+thisHumidity+"%  -->tulis log ke file
"+thisFile+".txt");
  Serial.println();

  //---------simpan log ke file--------------------
  myFile = SD.open(thisFile+".txt", FILE_WRITE);
    if (myFile) {
      myFile.print(thisDay+","+thisTime);
      myFile.print(","+thisTemp);
      myFile.print(","+thisHumidity);
//urutan log output = pemanas,pendingin,pelembab,pengering
      myFile.print(","+output);
      myFile.println();
      myFile.close();
      }
  }
}

void draw(void) {
  u8g.drawFrame(0,24,128,40);
  u8g.drawLine(64,24, 64,64);
  if (jalan<1){
    //tampilan program saat belum berjalan
  u8g.setFont(u8g_font_ncenR08);
  u8g.setPrintPos(11,20);
  u8g.print("--=>>  Stopped  <<=--");
  u8g.setFont(u8g_font_ncenR08);
  u8g.setPrintPos(74,36);
  u8g.print("          ");
  /*
  u8g.print("?? ~ ?? %);  */
  u8g.setFont(u8g_font_ncenR08);
  u8g.setPrintPos(10,36);
  /*
  u8g.print("?? ~ ?? 'C"); */
  u8g.print("          ");
  }
  else {
  //tampilkan hari ke dan jam ke
  u8g.setFont(u8g_font_ncenR08);
  u8g.setPrintPos(8,20);
  u8g.print("RUN:  ");
  u8g.setFont(u8g_font_profont12);
  u8g.print(hari);
  u8g.setFont(u8g_font_ncenR08);
  u8g.print(" Hari,  ");
  u8g.setFont(u8g_font_profont12);
  u8g.print(jam);
  u8g.setFont(u8g_font_ncenR08);
  u8g.print(" Jam");

  // tampilkan setpoin kelembaban berdasarkan tabel hari ke:
  u8g.setFont(u8g_font_ncenR08);
  u8g.setPrintPos(74,36);
  u8g.print(hd);
  u8g.print(" ~ ");
  u8g.print(hu);
  u8g.print(" %");

  // tampilkan setpoin suhu berdasarkan tabel hari ke:
  u8g.setFont(u8g_font_ncenR08);
  u8g.setPrintPos(10,36);
  u8g.print(cd);
  u8g.print(" ~ ");
  u8g.print(cu);
  u8g.print(" \260C");
  }

  //u8g.setFont(u8g_font_6x13);
  //u8g.setFont(u8g_font_gdr14);
  u8g.setFont(u8g_font_gdr20);
  // ukur suhu
  thisTemp = String(temperature,1); // + "\260C";
  const char* newTempC = (const char*) thisTemp.c_str();
  u8g.drawStr(8,58, newTempC);
 // ukur kelembaban
  thisHumidity = String(humidity,1); //+ "%";
  const char* newHumidity = (const char*) thisHumidity.c_str();
  u8g.drawStr(70,58, newHumidity);

 //***** RTC **********
  //DateTime now = RTC.now();
  // tampilan tanggal
  const char* newDay = (const char*) thisDay.c_str();
  u8g.setFont(u8g_font_ncenR08);
  u8g.drawStr(10,8, newDay);
  // ********************
  // tampilan jam
  const char* newTime = (const char*) thisTime.c_str();
  //const char* newTime = (const char*) thisTime1.c_str();
  u8g.setFont(u8g_font_profont12);
  u8g.drawStr(74,8, newTime);
}

void kirim_sms (void){
  jalan=EEPROM.read(0);
  if (jalan>0 ){
  Serial.println();
  Serial.println(thisDay+","+thisTime+", kirim SMS");
  Serial.println("Suhu "+thisTemp+"'C");
  Serial.println("Kelembaban "+thisHumidity+"%");
  Serial.println(thisAbntext);
  Serial.println();
  mySerial.println("AT+CMGF=1"); // Configuring TEXT mode
  delay(500);
  mySerial.println("AT+CMGS=\"+6285326858275\"");  //isi nomer HP
  delay(500);
  mySerial.print(hari);
  mySerial.print(" hari, ");
  mySerial.print(jam);
  mySerial.println(" jam");
  mySerial.println("Suhu "+thisTemp+"'C");
  mySerial.println("Kelembaban "+thisHumidity+"%");
  mySerial.println("Tgl. "+thisDay);
  mySerial.println("Jam "+thisTime);
  mySerial.println();
  mySerial.println(thisAbntext);
  mySerial.println();
  delay(200);
  mySerial.write(26);
      }
  thisAbntext="";
}

void jalan_program (void){
  EEPROM.update(0,1); //stat
  jalan=EEPROM.read(0);
  smspreviousMillis=millis();
  jampreviousMillis=millis();
  logpreviousMillis=millis();
  digitalWrite(buzzer,HIGH);
  delay(150);
  digitalWrite(buzzer,LOW);
}

void henti_program (void){
  digitalWrite(pelembab,LOW);
  digitalWrite(pengering,LOW);
  digitalWrite(pemanas,LOW);
  digitalWrite(pendingin,LOW);
  //reset eeprom
  EEPROM.update(0,0); //stat
  EEPROM.update(1,1); //hari
  EEPROM.update(2,0); //jam

  digitalWrite(buzzer,HIGH);
  delay(100);
  digitalWrite(buzzer,LOW);
  delay(50);
  digitalWrite(buzzer,HIGH);
  delay(100);
  digitalWrite(buzzer,LOW);
}
```

Lampiran 2: Tabel data suhu, kelembaban dan status output di hari ke-3 (data ke 1~25) dengan batas suhu acuan 27~30’C dan di hari ke-6 (data ke 26~50) dengan batas suhu acuan 25~28’C.

Data ke-	Suhu	Kelembaban	Output				Nilai Error Suhu	Persentase Error Suhu (%)	Nilai Error Kelembaban	Persentase Error Kelembaban (%)
1	30.80	67.70	0	1	0	0	0.8	2.67	0	0.00
2	30.30	74.90	0	1	0	1	0.3	1.00	4.9	7.00
3	29.30	79.20	0	0	0	1	0.0	0.00	9.2	13.14
4	29.00	78.80	0	0	0	1	0.0	0.00	8.8	12.57
5	28.70	77.20	0	0	0	1	0.0	0.00	7.2	10.29
6	28.60	76.70	0	0	0	1	0.0	0.00	6.7	9.57
7	28.70	76.40	0	0	0	1	0.0	0.00	6.4	9.14
8	28.80	75.70	0	0	0	1	0.0	0.00	5.7	8.14
9	28.90	75.20	0	0	0	1	0.0	0.00	5.2	7.43
10	28.90	74.60	0	0	0	1	0.0	0.00	4.6	6.57
11	29.00	74.30	0	0	0	1	0.0	0.00	4.3	6.14
12	29.10	73.90	0	0	0	1	0.0	0.00	3.9	5.57
13	29.10	73.40	0	0	0	1	0.0	0.00	3.4	4.86
14	29.20	73.00	0	0	0	1	0.0	0.00	3	4.29
15	29.20	72.60	0	0	0	1	0.0	0.00	2.6	3.71
16	29.20	72.50	0	0	0	1	0.0	0.00	2.5	3.57
17	29.30	72.50	0	0	0	1	0.0	0.00	2.5	3.57
18	29.30	72.60	0	0	0	1	0.0	0.00	2.6	3.71
19	29.30	72.30	0	0	0	1	0.0	0.00	2.3	3.29
20	29.40	71.30	0	0	0	1	0.0	0.00	1.3	1.86
21	29.40	71.20	0	0	0	1	0.0	0.00	1.2	1.71
22	29.40	71.20	0	0	0	1	0.0	0.00	1.2	1.71
23	29.50	71.60	0	0	0	1	0.0	0.00	1.6	2.29
24	29.50	71.30	0	0	0	1	0.0	0.00	1.3	1.86
25	29.50	71.10	0	0	0	1	0.0	0.00	1.1	1.57
26	28.80	83.30	0	0	0	1	0.8	2.86	13.3	19.00
27	28.50	88.10	0	0	0	1	0.5	1.79	18.1	25.86
28	28.60	82.80	0	0	0	1	0.6	2.14	12.8	18.29
29	28.70	79.80	0	0	0	1	0.7	2.50	9.8	14.00
30	28.80	77.10	0	0	0	1	0.8	2.86	7.1	10.14
31	28.90	75.50	0	0	0	1	0.9	3.21	5.5	7.86

Data ke-	Suhu	Kelembaban	Output				Nilai Error Suhu	Persentase Error Suhu (%)	Nilai Error Kelembaban	Persentase Error Kelembaban (%)
32	28.90	74.70	0	0	0	1	0.9	3.21	4.7	6.71
33	28.40	87.90	0	0	0	1	0.4	1.43	17.9	25.57
34	28.50	85.80	0	0	0	1	0.5	1.79	15.8	22.57
35	28.50	83.30	0	0	0	1	0.5	1.79	13.3	19.00
36	28.70	79.20	0	0	0	1	0.7	2.50	9.2	13.14
37	28.80	76.50	0	0	0	1	0.8	2.86	6.5	9.29
38	28.90	75.70	0	1	0	0	0.9	3.21	5.7	8.14
39	28.80	82.10	0	0	0	1	0.8	2.86	12.1	17.29
40	28.40	85.90	0	0	0	1	0.4	1.43	15.9	22.71
41	28.40	84.40	0	0	0	1	0.4	1.43	14.4	20.57
42	28.50	79.80	0	0	0	1	0.5	1.79	9.8	14.00
43	28.60	77.30	0	0	0	1	0.6	2.14	7.3	10.43
44	28.70	77.00	0	0	0	1	0.7	2.50	7	10.00
45	28.80	75.80	0	0	0	1	0.8	2.86	5.8	8.29
46	28.80	75.40	0	0	0	1	0.8	2.86	5.4	7.71
47	28.90	75.20	0	1	0	0	0.9	3.21	5.2	7.43
48	28.50	85.10	0	0	0	1	0.5	1.79	15.1	21.57
49	28.10	89.80	0	0	0	1	0.1	0.36	19.8	28.29
50	28.00	89.50	0	0	0	1	0.0	0.00	19.5	27.86
Rata - rata								1.18		10.59