```
/**********************************************
*  TITLE: Liquid level control using PID      *
*  Group members: Deepak Sethupathy MSM17B004 *
*                 Sriram S MSM17B016          *
*                                             *
***********************************************/
#include <LiquidCrystal.h>

LiquidCrystal lcd(2,3,4,5,6,7);  // initialize the LCD library by providing the
nuber of pins to it

// defines pins numbers
const int trigPin =8;
const int InputPin =9;

// defines variables
long duration;
float distance,distance1;

// defines variables for PID
unsigned long lastTime;
int
Input,Output,OutputPin=10,lastInput,Kp=10,Kd=7,Ki=6,SampleTime=100,outMax=170,ou
tMin=0;
double outputSum,sampleTimeInSec;
float Setpoint=5;



void setup() {

  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(InputPin, INPUT); // Sets the echoPin as an Input
  Serial.begin(9600); // Starts the serial communication
  lcd.begin(16,2);
  Serial.begin(9600);
  lastTime=millis();
  pinMode(InputPin,INPUT);
  pinMode(OutputPin,OUTPUT);
  lastInput=0;
  analogWrite(OutputPin,map(0,0,5,0,170));
  outputSum=0;
  sampleTimeInSec=SampleTime/1000;
  distance1=15; //Initial water level in cm
  Setpoint=map(Setpoint,-10,15,0,170); // mapping the set point

}

void loop() {

  unsigned long now = millis();
```

```
// Clears the trigPin
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(InputPin, HIGH);

// Calculating the distance
  distance= duration*0.034/2;



  distance=distance1-distance; //calculating difference between initial and 
current level



  Ki=Ki * sampleTimeInSec;
  Kd=Kd / sampleTimeInSec;

  lcd.setCursor(0,0);                  //LCD SCREEN DISPLAY
  lcd.print("WATER LEVEL=");
  lcd.print(distance);
  lcd.setCursor(0,1);
  lcd.print("SET LEVEL=5");


  unsigned long timeChange = (now - lastTime);
  if(timeChange>=SampleTime)
   {
      /*Compute all the working error variables*/

      double Input = map(distance,-10,15,0,170);
      double error = Setpoint - Input;
      double dInput = (Input - lastInput);
      outputSum+= (Ki * error);

      if(outputSum > outMax) outputSum= outMax;
      else if(outputSum < outMin) outputSum= outMin;

      double output;
      output = Kp * error;
      output += outputSum - Kd * dInput;


      if(output > outMax) output = outMax;
      else if(output < outMin) output = outMin;
      Serial.print(Setpoint);
      Serial.print(",");
      Serial.print(Input);
      Serial.print(",");
      Serial.println(output);

      analogWrite(OutputPin,output);
      lastInput = Input;
      lastTime = now;

   }
}
```