Task B

```
# include < Arduino.h >
# include < wire.h >
# include < Adafruit_GfX.h >
# include < Adafruit_SSD1306.h >
# define LED1 17
# define LED2 18
# define LED3 19
# define BTN 27
# define Buzzer 14
# define long_Press_MS 1500
# define screen_WIDTH 128
# define Screen_Height 64
  Adafruit_SSD1306
  OLED (Screen_WIDTH, SCREEN_HEIGHT, 2 Wi
int melody = {262, 294, 330, 349, 392, 440}
 bool melodyPlaying = false,
 void updateDisplay (const char * msg)
{
  Oled.clearDisplay();
  OLED.setTextsize(1);
  OLED.setTextcolor(SSD1306_White);
  OLED.setcursor(0,25);
  OLED.print(msg);
  OLED.display();
}
  void playmelody (){
  for (int i=0 ; i<8 && melodyPlaying;
        i++) {
   led c write Tone (0, melody[i]);
  delay(300);
}
}
```

```
void stopMelody(){
  ledcWriteTone(0,0);
  melodyPlaying = false;
}
void setup(){
  pinMode(LED1, output);
  pinMode(LED2, output);
  pinMode(LED3, output);
  pinMode(BTN, INPUT_PULLUP);
  ledcAttachPin(Buzzer, 0);
  wire.begin(21, 22);
  oled.begin(SSD1306_SWITCHCAPVC,
  updateDisplay("Ready");
  digitalWrite(LED1, low);
  digitalWrite(LED2, low);
  digitalWrite(LED3, low);
}
void loop(){
  if(digitalRead(BTN)==low){
  unsigned long startTime = millis();
  while(digitalRead(BTN)==low)
  delay(10);
  unsigned long PressTime = millis()-
  startTime;
  If(pressTime >= Long_Press_MS){
    melodyPlaying = true;
  updateDisplay("long Press Melody a);
}
  else{
    if(melodyPlaying){
      stopMelody();
    updateDisplay("Melody stopped
```

```
    }
  } else {
    static bool led state = fale;
    led state = ! led state
    digital write ( led 1 , led state);
    digital write ( led 2 , led state);
    digital write ( led 3 . led state);
    update Display ("Short Press : LED");
    }
}
```