src\main.c

```
// Practice assignment 4

#include <stdio.h>
#include <avr/io.h>
#include <util/delay.h>
#include <avr/eeprom.h>

#include "i2cmaster.h"
#include "lcd.h" //library init
#include "lm75.h"

int main(void) {

  i2c_init(); // initialize I2C and LCD
  LCD_init();

  DDRC = 0xF0; // set data direction for port C pins, 0-3 as input (i.e. the buttons)
  PORTC = 0x3F; // set pull-up resistor for port C
  DDRD = 0xFF; // set data direction for port D, all output
  PORTD= 0x00; // set output for port D (none)

  float currTemp,maxTemp,minTemp; // create variables for temps
  unsigned int EEPROMAddr=0; // variable for address

  EEPROMAddr=0; //set addr for the thing to write to
  maxTemp=eeprom_read_float((float *)EEPROMAddr); //write max temp
  EEPROMAddr=2; //again but 2 bytes later (bc float is 16-bit wide)
  minTemp=eeprom_read_float((float *)EEPROMAddr); //write min temp

  while(1) { // start program loop

    if(!((PINC>>3)& 1)) { //read button DI3
      minTemp=125; // set min temp to maximum temp of LM75, if pressed
      maxTemp=-55; // backwards
    }
    currTemp=get_temperature(); // get temperature from LM75

    if(currTemp>maxTemp) {// set max temp if the current temp is greater than it, store it in
EEPROM at addr 0
      maxTemp=currTemp;
      EEPROMAddr=0;
      eeprom_write_float((float *)EEPROMAddr, maxTemp);
    }

    if(currTemp<minTemp){ // backwards
      minTemp=currTemp;
      EEPROMAddr=2;
      eeprom_write_float((float *)EEPROMAddr, minTemp);
```

```
48      }
49
50      LCD_set_cursor(0,0);
51      printf("Current temp: %3.2fC",currTemp); // print current temp in Celsius
52
53      LCD_set_cursor(0,1);
54      printf("Max temp: %3.2fC",maxTemp); //print maximum temp in Celsius
55
56      LCD_set_cursor(0,2);
57      printf("Min temp: %3.2fC",minTemp); // same but minimum
58
59      _delay_ms(950);
60    }
61
62    return 0;
63  }
64
```