Class-3, Task Program LCD

```c
#define _XTAL_FREQ 20000000  // Define the crystal oscillator frequency as 20MHz (for delay funct
void Lcdinit(void);          // Function prototype for LCD initialization
void LcdCommand(uint8_t);    // Function prototype for sending commands to the LCD
void LcdData(uint8_t);       // Function prototype for sending data to the LCD
uint8_t x, i, m;             // Declare global variable
uint8_t name[18] = {"SAGAR"}; // Character array storing the name to display

void main(void){
    Lcdinit();  // Initialize the LCD
    // Display "SAGAR" starting from the first row, first column (0x80)
    for (uint8_t i = 0; i < 11; i++){
        LcdCommand((uint8_t)(0x82 + i));  // Move the cursor to the respective position (casting
        LcdData(name[i]);                 // Send each character to the LCD
    }

    // Display "SAGAR" starting from the second row, last column (0xCF)
    for (uint8_t i = 0; i < 11; i++){
        LcdCommand((uint8_t)(0xCD - i));  // Move cursor from right to left on the second row (ca
        LcdData(name[i]);                 // Send each character to the LCD
    }
    while(1);  // Infinite loop to keep the program running
}
void Lcdinit(void) {
    TRISC = 0x00;  // Set PORTC as output (for control signals)
    TRISD = 0x00;  // Set PORTD as output (for data signals)

    __delay_ms(100);  // Wait for LCD to stabilize

    // LCD initialization sequence as per the HD44780 LCD datasheet
    LcdCommand(0x30);  // Send function set command (8-bit mode)
    __delay_ms(100);   // Delay for command execution
    LcdCommand(0x30);  // Repeat function set command
    __delay_ms(100);
    LcdCommand(0x30);  // Repeat function set command again
    __delay_ms(100);
    LcdCommand(0x38);  // Set LCD for 8-bit mode, 2-line display, 5x8 font
    __delay_ms(100);
    LcdCommand(0x0C);  // Turn on display, cursor off
    __delay_ms(100);
    LcdCommand(0x01);  // Clear the display
    __delay_ms(100);
}
// Function to send data (characters) to the LCD
void LcdData(uint8_t i) {
    PORTC |= (0x1 << 3);  // Set RS (RC3) = 1 (indicates data mode)
    PORTD = i;            // Place data on PORTD
    PORTC |= (0x1 << 0);  // Set EN (RC0) = 1 (enable pulse start)
    __delay_ms(100);      // Small delay for command execution
    PORTC &= ~(0x1 << 0); // Set EN (RC0) = 0 (enable pulse end)
}

// Function to send commands to the LCD
void LcdCommand(uint8_t i) {
    PORTC &= ~(0x1 << 3);  // Set RS (RC3) = 0 (indicates command mode)
    PORTD = i;             // Place command on PORTD
    PORTC |= (0x1 << 0);   // Set EN (RC0) = 1 (enable pulse start)
    __delay_ms(100);       // Small delay for command execution
    PORTC &= ~(0x1 << 0);  // Set EN (RC0) = 0 (enable pulse end)
}
```