Daniel:

char \*message = "Hello World - April 2016"; // Store message for line 2 in a variable

for(int i = 0; i < 9; i++) // For loop to add to the message pointer

{

LCDwriteLineTwo((message+i)); // Print message starting at incremented pointer up to \0

\_\_delay\_ms(1000); // Delay cannot be much longer than a second due to program limits

\_\_delay\_ms(1000);

\_\_delay\_ms(250);

}

\_\_delay\_ms(1000); // Wait 1 extra second before beginning again

Shawn:

TRISBbits.RB5 = 1; // RB5 is button input

WPUBbits.WPUB5 = 1; // Weak pullup enabled for B5

\_\_delay\_ms(100); // Wait for things to settle

while(!PORTBbits.RB5) // Wait until B5 goes low

{

continue;

}

Kyle:

#include "lcd.h" // Add LCD library

// LCD initilization

LCDinit();

LCDputs("Ruth"); // Add Ruth to line 1

James:

/// CONFIG1H

#pragma config FOSC = INTIO67 // Oscillator Selection bits (Internal oscillator block)

#pragma config PLLCFG = OFF // 4X PLL Enable (Oscillator used directly)

#pragma config PRICLKEN = ON // Primary clock enable bit (Primary clock enabled)

#pragma config FCMEN = OFF // Fail-Safe Clock Monitor Enable bit (Fail-Safe Clock Monitor disabled)

#pragma config IESO = OFF // Internal/External Oscillator Switchover bit (Oscillator Switchover mode disabled)

// CONFIG2L

#pragma config PWRTEN = OFF // Power-up Timer Enable bit (Power up timer disabled)

#pragma config BOREN = OFF // Brown-out Reset Enable bits (Brown-out Reset disabled in hardware and software)

#pragma config BORV = 190 // Brown Out Reset Voltage bits (VBOR set to 1.90 V nominal)

// CONFIG2H

#pragma config WDTEN = OFF // Watchdog Timer Enable bits (Watch dog timer is always disabled. SWDTEN has no effect.)

#pragma config WDTPS = 32768 // Watchdog Timer Postscale Select bits (1:32768)

// CONFIG3H

#pragma config CCP2MX = PORTC1 // CCP2 MUX bit (CCP2 input/output is multiplexed with RC1)

#pragma config PBADEN = ON // PORTB A/D Enable bit (PORTB<5:0> pins are configured as analog input channels on Reset)

#pragma config CCP3MX = PORTB5 // P3A/CCP3 Mux bit (P3A/CCP3 input/output is multiplexed with RB5)

#pragma config HFOFST = OFF // HFINTOSC Fast Start-up (HFINTOSC output and ready status are delayed by the oscillator stable status)

#pragma config T3CMX = PORTC0 // Timer3 Clock input mux bit (T3CKI is on RC0)

#pragma config P2BMX = PORTD2 // ECCP2 B output mux bit (P2B is on RD2)

#pragma config MCLRE = INTMCLR // MCLR Pin Enable bit (RE3 input pin enabled; MCLR disabled)

// CONFIG4L

#pragma config STVREN = ON // Stack Full/Underflow Reset Enable bit (Stack full/underflow will cause Reset)

#pragma config LVP = OFF // Single-Supply ICSP Enable bit (Single-Supply ICSP disabled)

#pragma config XINST = OFF // Extended Instruction Set Enable bit (Instruction set extension and Indexed Addressing mode disabled (Legacy mode))

// CONFIG5L

#pragma config CP0 = OFF // Code Protection Block 0 (Block 0 (000800-003FFFh) not code-protected)

#pragma config CP1 = OFF // Code Protection Block 1 (Block 1 (004000-007FFFh) not code-protected)

#pragma config CP2 = OFF // Code Protection Block 2 (Block 2 (008000-00BFFFh) not code-protected)

#pragma config CP3 = OFF // Code Protection Block 3 (Block 3 (00C000-00FFFFh) not code-protected)

// CONFIG5H

#pragma config CPB = OFF // Boot Block Code Protection bit (Boot block (000000-0007FFh) not code-protected)

#pragma config CPD = OFF // Data EEPROM Code Protection bit (Data EEPROM not code-protected)

// CONFIG6L

#pragma config WRT0 = OFF // Write Protection Block 0 (Block 0 (000800-003FFFh) not write-protected)

#pragma config WRT1 = OFF // Write Protection Block 1 (Block 1 (004000-007FFFh) not write-protected)

#pragma config WRT2 = OFF // Write Protection Block 2 (Block 2 (008000-00BFFFh) not write-protected)

#pragma config WRT3 = OFF // Write Protection Block 3 (Block 3 (00C000-00FFFFh) not write-protected)

// CONFIG6H

#pragma config WRTC = OFF // Configuration Register Write Protection bit (Configuration registers (300000-3000FFh) not write-protected)

#pragma config WRTB = OFF // Boot Block Write Protection bit (Boot Block (000000-0007FFh) not write-protected)

#pragma config WRTD = OFF // Data EEPROM Write Protection bit (Data EEPROM not write-protected)

// CONFIG7L

#pragma config EBTR0 = OFF // Table Read Protection Block 0 (Block 0 (000800-003FFFh) not protected from table reads executed in other blocks)

#pragma config EBTR1 = OFF // Table Read Protection Block 1 (Block 1 (004000-007FFFh) not protected from table reads executed in other blocks)

#pragma config EBTR2 = OFF // Table Read Protection Block 2 (Block 2 (008000-00BFFFh) not protected from table reads executed in other blocks)

#pragma config EBTR3 = OFF // Table Read Protection Block 3 (Block 3 (00C000-00FFFFh) not protected from table reads executed in other blocks)

// CONFIG7H

#pragma config EBTRB = OFF // Boot Block Table Read Protection bit (Boot Block (000000-0007FFh) not protected from table reads executed in other blocks)