```
/*************************************************************/
//  Project:    Lab 3
//
//  Name:   Quang Nguyen
//  Date:   1/31/2014
//
//  Function:
//          1)  Using a loop that will index the PORT A single output that will be true,
//      write code that will set each of the lower 8 bits in PORT A to true one at a time.
//          2)  The bit will stay true for 1 second then turn off and the next bit
//      higher will turn on immediately and stay on for 1 second and so on.
//          3)  When the PORT A bit 7 is true for one second and turns off PORTA bit 0 will turn on.

//  Pins used:
//          Lower 8 bits of PORTA
//
//  Peripherals used:
//          * Timer 1 used for the delay loop
//                  TCKPS (Prescaler) = 256
//                  1/16 MHz = 6.25 E-8
//                  6.25 E-8 * 256 = 0.000016 or 1.6 E-5
//                  6250 instructions * .000016 = 0.1 second ***
//                  We are going to let this overflow 10 times to make 1 second
//
//  Comments:
//
/*************************************************************/

/*************** Configuration Bits **************************/
#include <xc.h>

// FOSCSEL
#pragma config FNOSC = PRIPLL           // Oscillator Mode (Primary Oscillator (XT, HS, EC) w/ PLL)
#pragma config IESO = ON                // Two-speed Oscillator Start-Up Enable (Start up with FRC,
then switch)

// FOSC
#pragma config POSCMD = XT              // Primary Oscillator Source (XT Oscillator Mode)
#pragma config OSCIOFNC = OFF           // OSC2 Pin Function (OSC2 pin has clock out function)
#pragma config FCKSM = CSDCMD           // Clock Switching and Monitor (Both Clock Switching and
Fail-Safe Clock Monitor are disabled)

// FWDT
#pragma config FWDTEN = OFF             // Watchdog Timer Enable (Watchdog timer enabled/disabled by
user software)

/*************************************************************/

/*************** Library includes ****************************/
#include <p33FJ256GP710A.h>

/*************** Constants Declaration ***********************/

/*************** Function Prototype **************************/
void initialize();

/*************** Main Program ********************************/
int main() {
    // setting up everything
    initialize();
    int Timer1Counter = 0;

    /*************** Main Loop *********************************/
    while (1) {
        // Turn first bit of PORT A ON
        PORTA = 1;
        // For loop to turn on PORT A one at a time
        int i;
        for (i = 0; i < 8; i++)
        {
            // Delay 1 sec
```

```
            Timer1Counter = 0;
            while (Timer1Counter < 10)
            {
                if (TMR1 == 6250)
                {
                    Timer1Counter++;
                    TMR1 = 0;
                }
            }       // End of Timer1Counter while loop
            // Bit shift left by 1
            PORTA = PORTA << 1;
        }// End of FOR loop

    } // End of the infinite While loop

    return 1;
}

/*************** Initialize Function ************************/
void initialize() {
    /******* Setting up for Clock (PLL, M, N1, N2) for 32 MHz and Fcy = 16 MHz *****/
    // Fosc = Fin(M/(N1*N2)) = 8 MHz (32/(2*4)) = 32 MHz
    PLLFBD = 30; // M = 32
    // N1 default is 2
    // N2 default is 4
    // Fcy = Fosc/2 by default
    /*******************************************************************************/

    // Setting up PORT A
    AD1PCFGH = 0xFF;                // Turn off ADC for Module 1
    TRISA = 0x00;                   // Port A to output

    // Setting up Timer 1 module
    T1CON = 0b0100000000110000;
            // TON = OFF
            // TGATE = OFF
            // TCKPS (Prescaler) = 256
                // 1/16 MHz = 6.25 E-8
                // 6.25 E-8 * 256 = 0.000016 or 1.6 E-5
                // 6250 instructions * .000016 = 0.1 second ***
                // We are going to let this overflow 10 times to make 1 second
            //  TSYNC = OFF
            // TCS = Internal Clock
    // Start Timer 1
    T1CONbits.TON = 1;

}
```