3.4) LATFbits.LATF0 = 1; from LATFbits.LATF0 = 0;
This should make it so both pins are either on or off at the same time

3.5)
```
#include <xc.h>          // Load the proper header for the processor

#define Maxcycles  1000000;
#define Deltacycles 100000;

void delay(cycles);

int main(void) {
  int cycle_m = Maxcycles;
  int cycle_d = Deltacycles;
  TRISF = 0xFFFC;        // Pins 0 and 1 of Port F are LED1 and LED2.  Clear
                         // bits 0 and 1 to zero, for output.  Others are inputs.
  LATFbits.LATF0 = 0;    // Turn LED1 on and LED2 off.  These pins sink current
  LATFbits.LATF1 = 1;    // on the NU32, so "high" (1) = "off" and "low" (0) = "on"

  while(1) {
    delay(cycle_m);
    LATFINV = 0x0003;    // toggle LED1 and LED2; same as LATFINV = 0x3;
    cycle_m-=cycle_d;
    if (cycle_m <= 0){
      cycle_m= Maxcycles;
    }
  }
  return 0;
}

void delay(cycles) {
  int j;
  for (j = 0; j < cycles; j++) { // number is 1 million
    while(!PORTDbits.RD7) {
        ;   // Pin D7 is the USER switch, low (FALSE) if pressed.
    }
  }
}
```

4.1)
- they are all global

4.2)
- code turned in. invest.c

- code turned in. main.c, helper.c and helper.h.
The main function was put in the main.c. The function prototypes and global variables were put in helper.h and the helper function definitions were put in helper.c.

- Only the main was put in main.c
- In calculate.c the function def for calculateGrowth was added with the prototype in calculate.h
- In io.c the rest of the function definitions were added with their prototypes and the investment struct in io.h

This was done mostly because of the question, all files included io.h but only main included calculate.h because io.c did not use the function.

4.4)
```
void LCD_ClearLine(int ln) {
	LCD_Move(ln, 0);
	const char msg[16] = "                ";
	LCD_WriteString(msg);
```