1. (2 pts) Write a program in DEBUG that does the four additions in HW2 Question 5: "Assuming an 8-bit representation, list 4 pairs of bit strings whose binary interpretations as unsigned and 2's complement numbers have the following properties: a) Both signed and unsigned addition is correct; b) Signed addition is correct, but unsigned is not; c) Unsigned addition is correct, but signed is not; d) Neither signed nor unsigned addition is correct." Since incorrect unsigned addition is represented by the Carry Flag set (CY) and incorrect signed addition is represented by the Overflow flag set (OV) you will be able to see these flags as you trace the program in Debug. Trace the program and submit a screen dump of the Debug trace showing the flags in each of the four cases, as well as a screen dump showing your program with the U command.

Example: correct for signed and unsigned interpretations:
```
mov al, 1
add al, 1
```

2. (1 pt) Write a short program consisting of a single MOV instruction followed by a series of 4 ADD instructions that will set each of the flags in this order: OF, CF, ZF, SF. In other words the first ADD should set OF and you don't have to worry about CF, SF, ZF. The second ADD should set CF, etc. Trace the program and submit a screen dump of the Debug trace showing the flags in each of the four cases, as well as a screen dump showing your program with the U command.

3. (3 pts) Write and assemble a program, using the data definitions below, that will convert a string to sentence case (first letter capitalized, all others lower case). Use function 9 to display the string to the console before and after processing the string.

   ```
   astr db 'this sentence HAS MiXed CASE'
   slen equ $-astr
   db '$'
   ```

Use Boolean operations with masking to perform the case conversion. You can process the first character and then do the following:
   a. Load a pointer to the second character in the string into an index register (e.g., si)
   b. Put character count in CX and decrement CX
   c. Convert a character (load, mask, and store)
   d. Loop until done

Submit source code and a screen dump showing output.

4. (4 pts) Adapt disptim1.a86 (the version with a bin2dec routine) to display the current date rather than the current time, using function 2Ah (Get Date). Here is the definition of int 21h function 2Ah from the OS function handout:

   **FUNCTION 2AH: Get System Date**  Returns the system date

   **Parameters:** NONE
   **Returns:**   AL Day of Week (Sunday = 0)   CX Year (1980-2099)
                 DH Month (1-12)   DL Day (1-31)

   Note that the year is a 16-bit integer, unlike the month and day. (Hint: look at the parameters used by bin2dec). Also modify the program to show your name. Format the date in ISO8601 date format (YYYY-MM-DD, for example 2010-03-18)

Submit your source code and a screen dump showing the formatted date in memory.