Program Set #2
Total Points: 30

Three problems must be implemented for full credit. The required (starred) problem marked in the set must be implemented by everyone. See the 2425 Grading and Submission Guide Sheets for additional grading/submission information. Partial credit will be given. (10 points each)

Note: If coding in C++, use the STL string class for problems involving strings. Do not use C style strings.

Section One- Choose two problems.

1. Write a program that converts a rational value less than 1 into hexadecimal form. Input from the keyboard a positive rational (decimal) number. Assume proper input. Output to the screen the value converted to hexadecimal using the following rules:
   - If the conversion includes a repeating decimal, output only one section of the repeating part. For example, output $.101101101..._{16}$ as `.101`
   - If the converted decimal does not repeat within the first 15 digits only output the first 10 digits
   - Do not output any repeating zeros at the end of a hex value.
   - Do not output the integer part of a number (`0.8` verses `.8`)

Finally, the program should ask if the user wants to run the program again (Check case). Refer to the sample output below.

Sample Run:

```
Enter a decimal integer: .65

Hexadecimal value: .A6

Run again (Y/N): y

Enter a decimal integer: .144

Hexadecimal value: .24DD2F1A9F

Run again (Y/N): N
```

Name the program: `HexNoConversionXX.cpp` or `HexNoConversionXX.java`, where `XX` are your initials.

2. Write a MASM program that outputs your first and last name to the screen. Output should be user friendly.

Name the program: `YourNameXX.asm`, where `XX` are your initials.

3. Write a MASM program that finds the greatest common divisor (GCD) between two numbers. The user will enter two integers from the keyboard and output to the screen the GCD of the two numbers. Assume proper input. Output should be user friendly.

Name the program: `GCDXX.asm`, where XX are your initials.

Required Problem- Comprehensive.

4 (**). Write a MASM program that finds the two's complement of a number (+/-). The user will enter a single integer from the keyboard and output to the screen the two's complement value of the number entered. Assume proper input. Output should be user friendly.

Name the program: `ComplementXX.asm`, where XX are your initials.

Extra Credit: Implement the following problem. See the 2425 Grading and Submission Guide Sheets for additional grading/submission information. Partial credit will be given. (10 points)

Write a MASM program that sorts a programmer-defined array. The user will enter 5 integer values from the keyboard. Do not assume the values will be entered in sorted order. Output to the screen the sorted array.

Name the program: `ArraySortXX.asm`, where XX are your initials.