Cs352 — Homework #2

Due Time: 2/10/04 (9:00PM)

Turnin ID: *cs352_assg2*

Turnin files: SumOfSquares.c, sum.c, MyReverse.c, MySubStringFinder.c.

(PS: For example, you may turnin multiple files by UNIX command “turnin *cs352_assg2* yourfile1 yourfile2 yourfile3 yourfile4”. You may also turnin one file at a time by “turnin *cs352_assg2* *file_you_want_to_turnin*”. Later turnin file will overwrite the old file which has the same filename. So if you turnin the same file multiple times, we only receive the last version you turned in. To see a list of the files you turned in, you may use the command “turnin -ls *cs352_assg2*”. For help information about turnin program, please use the command “turnin -h”. If you still have questions about turnin, please either stop by TA’s office hours or email TA’s.

Your code should follow the instructions in the “C coding guidelines”. In particular, pay attention to proper documentations)

We will put the sample executables in the class home dir at /home/cs352/spring04/sample_exec/assg2 on lectura. The purpose of these samples is to give you an idea how the expected solution look like (how it runs, how it gets input, how it format the output, etc). Please check your output format against the output generated by the sample executables with ”diff” command. If you found any bugs with the sample executables or have questions on them, please contact us.

1. Write a **recursive** function called `sum1`, whose prototype is `int sum1(int n)`, that received a integer n, and returns the value of the sum $1 + 4 + 9 + \ldots + n^2$. Write a program called `SumOfSquares.c` at which the number n is defined as a macro (with the value 17) that calls the function `sum1` and prints the values of the sum.

2. Write a C program (named `sum.c`) that reads (by scanf()) a list of integers from the stdin and print (by printf()) to the stdout the number of integers, the minimum integer, the maximum integer and the sum of all the integers. The input integers are separated by spaces and/or newlines. A sample run of the program (where ”sum” is the executable compiled from sum.c) is below:

   ```
   lec:xxx > sum
   1 2 3
   ```

```
4 5 6 7 8
8, 1, 8, 36
lec:xxx >
```

In the above example, 1 through 8 are the input numbers. ”8, 1, 8, 36” are number of integers, min, max and sum respectively. You may assume the input are all valid integers of type “int”. When there is no integer comming from stdin at all, your program should print “0, N/A, N/A, 0”.

The macros INT_MIN and INT_MAX in limits.h containing the maximum and minimum possible value of an integer might help.

3. Create a recursive function named `MyReverse` that recievse a pointer to a string, and prints the character in this string in a reverse order. So for example, is *char s*[] = “*Hello*″;, then the command *MyReverse*(*s*) should print *olleH*. Submit a file named MyReverse.c, that contains this function and a C program that calls this function.

4. Write a function called `MySubStringFinder`, whose prototype is `int MySubStringFinder(char *s1, char *s2)`. If *s2* appears as a substring in *s1*, then `MySubStringFinder` should return 1. Otherwise, it should return 0. For example

   ```
   char s1[]="HelloWorldWhatANiceDay" ;
   char s2[]="ldWha" ;
   ```

   Then `int MySubStringFinder( s1, s2)` should return 1; Submit a file named MySubStringFinder.c, that contains this function and a C program that calls this function.