

Computer Science 210 –Computer Organization

Homework Exercise 9

Due on Github 11:59 PM Monday 4 April

Your turn in will consist of some C language source files and a **makefile**.

Warmup: You will use the file **testnumbers.c** in the zipfile. This file requires you to build a library, so do not attempt to compile it yet.

In the rest of this exercise, you will build a library of functions to process arrays of integers. These include functions for terminal I/O, computing the sum, and computing the average.

1. Read the file **testnumbers.c**. This is a short test driver program for the library. Study the code and note the header file name and the use of the array processing functions. Here is a trace of two runs of this program:

```
$ make
gcc -c testnumbers.c
gcc -c numbers.c
gcc -o testnumbers testnumbers.o numbers.o
$ ./testnumbers
Enter a number or 0 to stop: 4
Enter a number or 0 to stop: 8
Enter a number or 0 to stop: 0
```

```
The input numbers are
4
8
The sum is 12
The average is 6.000000
$ ./testnumbers
Enter a number or 0 to stop: 1
Enter a number or 0 to stop: 2
Enter a number or 0 to stop: 3
Enter a number or 0 to stop: 66
Enter a number or 0 to stop: 22
Maximum number of numbers entered
```

```
The input numbers are
1
2
3
66
22
```

```
The sum is 94
The average is 18.800000
$ ./testnumbers
Enter a number or 0 to stop: 0
```

No numbers were entered

2. Write the header file for the library (**numbers.h**). This file should begin with a comment that includes your name and the purpose of the file.
3. Write the implementation file for the library (**numbers.c**). Again, include a prefatory comment. This file should include the complete definitions of the array processing functions.
4. Compile and link the files via the command prompt and test until you see no errors.
5. Now ensure you only have the **.h** and **.c** files in your directory (*be very careful on this one!*). Lastly, create a **makefile** for your software and run the **make** command to rebuild your application.