

Name: \_\_\_\_\_

Problem	Points	Score
1	100	
Total	100	

Notes:

- (1) For this exam you are allowed to open a terminal window on your computer, you are allowed to web surf with Google, but you cannot use online chat or other interactive services.
- (2) The first step in this exam is to create a workspace in the following directory:  
`/data/courses/ece_1111/current/exams/ex_03/lastname_firstname`  
Put all your code here.
- (3) Set the permissions using “`chmod -R u+rwx,g-rwx,o-rwx <lastname_firstname>`” so only you have read and write permission to this directory.

In this directory:

```
ece-000_[2]: p
/data/courses/ece_1111/resources/data/binary
ece-000_[2]: d
total 12
drwxrwxr-x. 2 picone ece_1111 51 Apr 21 23:19 ./
drwxr-xr-x. 6 picone ece_1111 73 Apr 21 23:15 ../
-rw-rw-r--. 1 picone ece_1111 20 Apr 21 23:18 f00.dat
-rw-rw-r--. 1 picone ece_1111 20 Apr 21 23:19 f01.dat
-rw-rw-r--. 1 picone ece_1111 18 Apr 21 23:19 f02.dat
```

you will find some files containing binary data – 16-bit short integers. Use “od -s” to view the data.

Write a C++ program that computes a histogram of the numbers contained in these files. Your program should function like this:

```
ece-000_[2]:p01.exe <filenames>
0: 3
1: 5
2: 33
...
99: 1
```

Assume the numbers are between [0, 100]. Only print out a histogram value if the number of times it occurred is nonzero. Your program must run from any directory, so assume the full pathnames of the files are given.

You must implement this using four files: Makefile, header file (p01.h), driver program (p01.cc), implementation file (p01\_00.cc). Create a class called MyHistogram that has the following methods:

```
MyHistogram::init();
MyHistogram::add(char* file);
MyHistogram::print(FILE* fp);
```

The method `init()` initializes the counts to zero. The method `add()` opens the file, reads the samples, and adds them to the histogram. The method `print()` prints the histogram in the format above.

The driver program should loop over the command line arguments (multiple files) and call the `add()` method. Once all the files have been processed, it should print the histogram and exit.

The output for the above files should be:

```
ece-000_[2]:p01.exe <filenames>
10: 1
11: 1
12: 1
...
30: 1
31: 1
...
34: 1
36: 1
...
39: 1
```

Your program should work for any set of files. Try repeating a file twice and see if the counts double.