Name: _____

Problem	Points	Score
1	50	
2	50	
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_02/lastname_firstname

(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. Create two subdirectories within this directory: p01 and p02. You will use these for problems 1 and 2 respectively. Put ALL your code in these directories. Do not touch your files after the exam is over.

(4) You must use a make file, a header file and a main program file named p01.cc (or p02.cc). All other code needs to go into an implementation file called p01_00.cc (or p02_00.cc).

Problem No. 1: This binary file:

/data/courses/ece_1111/current/exams/ex_02/picone_joseph/p01.bin

contains a total of 19 bytes. The first 5 bytes are unknown. The next 10 bytes are the value "27.0" written in ASCII using an fprintf statement with a format of "%10.4f". The last 4 bytes are unknown.

Decode the unknown bytes using the od command. In the file p01/OD.TXT, place the sequence of od commands used to identify what data types were written to this file. If you correctly decode the bytes, the corresponding values will be "27" or "27.0".

You must complete this step before you can proceed to the second part of this problem.

Next, write a C program to correctly read and print the values in this file. Your interface and output should look like this:

```
ece-000_[1]: p01.exe /data/courses/ece_1111/current/exams/ex_02/picone_joseph/p01.bin
1: <type> value
2: <type> value

…
```

For the ASCII data, use a formatted print statement with "<type>" replaced with "<ASCII>".

Note that your program must work for any values contained in the file assuming the sequence of data types is the same. You have to actually read and interpret the bytes. I will test your program with a file in which the value 27 is replaced by some arbitrary value.

Problem No. 2: In HW #7, we learned how to read a file using frames and windows. Modify your program so that it reads a binary file of short integers using a frame size of M samples and a window size of W samples. Your interface must be the following:

p02.exe <filename> M W

Use this file for testing:

/data/courses/ece_1111/current/exams/exam_02/picone/p02.bin

This file contains short integers ranging in value from 1 to 20:

```
nedc027_[1]: od -s /data/courses/ece_1111/current/exams/ex_02/picone_joseph/p02.bin
0000000         1       2       3       4       5       6       7       8
0000020         9      10      11      12      13      14      15      16
0000040        17      18      19      20
0000050
```

There are two things your program must do differently:

(1) The window, instead of being center-aligned, must be left-aligned. This means the first window should be the first W samples in the file. The second window must be this window shifted by M samples. The third window must be the second window shifted by M samples. For example, for p02.bin for M =2 and W = 4:

```
frame 1: [1 2 3 4]
frame 2: [3 4 5 6]
frame 3: [5 6 7 8]
```

Your code must work for all combinations of M and W and different file lengths, and must handle the end of file condition properly (values beyond the end of file are assumed to be zero). You must handle the case where W < M.

You must only read M samples with each iteration after initialization. You cannot simply position the file pointer and read W samples each time. The total number of samples you read must be equal to M so that you are not doing unnecessary I/O.

(2) You must multiply each window of data, s_w, and a weighting function, h, where h is a vector of length W whose values are all 2. Then you must sum the squares of the result to compute "energy". Print the following information to stdout: input filename, M, W, and for each frame, the frame index and the energy value.

In the previous example, the output will be (after the weighing function is applied):

```
M =2 and W = 4:
  frame 1: 120
  frame 2: 344
```

$[(\mathbf{1}*\mathbf{2})^2+(\mathbf{2}*\mathbf{2})^2+(\mathbf{3}*\mathbf{2})^2+(\mathbf{4}*\mathbf{2})^2\,]$

$[(\mathbf{3}*\mathbf{2})^2+(\mathbf{4}*\mathbf{2})^2+(\mathbf{5}*\mathbf{2})^2+(\mathbf{6}*\mathbf{2})^2\,]\ldots$