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.

Problem No. 1: Write a program to decode a binary file. The specification for the file is as follows:

- The first three bytes contain a version number – either "v00" or "v01"
- If the first three bytes are "v00", the next byte is a character, followed by a short integer, followed by a 4-byte float
- If the first three bytes are "v01", the next four bytes are a floating-point number.

Use the attached file, *exam_02_p1.dat*, as a test case to debug your code. It contains the character string "v00", the character "a", the short integer 27, and the floating-point number 27.2727.

Your program must follow this interface;

p1.exe <filename>

and must print the values it finds to stdout. You must use fopen, fclose, fread and fprintf. You cannot use cin or other such libraries (there is no need to use these).

Submit **p1.cc** as your solution to this problem. Make sure your code is well-commented and formatted, or you will not receive partial credit.

Problem 2: Create a modular structured program that consists of the following files: **Makefile, p2.h, p2.cc and p2_00.cc**. The make file will compile and link the binary p2.exe.

The main program should use this interface:

p2.exe step_size

Your program should generate samples of a linear function that ranges from [0,1] in steps of "step_size". For example, if step_size = 0.25, your program would output the numbers 0.00, 0.25, 0.5, 0.75 and 1.0.

The main program must call three functions:

boolean p2_allocate(...);

boolean p2_compute(float* values, long N);

boolean p2_deallocate(...);

The function *p2_allocate()* creates space for an array of N samples. The function *p2_compute()* fills the array with the proper values. The function *p2_deallocate()* frees up space that was allocated in the allocate function. You need to decide what arguments are necessary for *p2_allocate()* and *p2_deallocate()* to make this program work.

Print the values of the array to stdout using a nicely formatted fprintf statement. Submit the above files (well commented and formatted of course) as an email attachment using the exact names above.

Summary: You will submit the files p1.cc, Makefile, p2.h, p2.cc and p2_00.cc as attachments to your email as your solution to this exam.