**ECE 1111: Engineering Computation I**

**Laboratory No. 13: Binary Data**

**Goals:** Help you develop an understanding of how to do binary I/O and basic signal processing in C++ and Python.

**Deposit your work in:**

/data/courses/ece\_1111/current/labs/lab\_13/<lastname\_firstname>

**Description:**

1. (/p01) In C++, generate a binary file that contains *64,000* samples of a *100* Hz sinewave. Write short integer values to the file. Use a sample frequency of *8,000* Hz and have the amplitudes range from [-*10,000*, *10,000*]. This means the file will contain *8* seconds of a signal consisting of a sinewave where each sample of the sinewave represents the value of the sinewave at an increment of *1/8000* secs. The calculation you want to implement is:

where *i* ranges from *0* to *64,000*. The file should have a size of *64,000* samples \* *2* bytes/sample = *128,000* bytes.

1. (/p02) Write a Python program that does the same thing. Using the Unix diff command, demonstrate that the two files these programs created are identical. If there are discrepancies, explain why.
2. (/p03) Using what you did in HW #9, write a C++ program that computes the energy of the signal. You should be able to specify the frame and window sizes from the command line:

p03.exe <file> frame\_size window\_size

The energy in the window is defined as the sum of the squares:

1. (/p04) Repeat no. 3 in Python and show that you get the same results.

**Summary:**

Python makes it very easy to do many things, but when you get down into the details of various signal processing algorithms, there is not much difference between the two languages, because a lot of the effort is devoted to what we call “infrastructure” – organizing your data so you can do computations on it.