Name:

Do your work in this directory:

/data/courses/ece\_1111/current/quizzes/qu\_06/lastname\_firstname/p01

Make sure your parent directory has the correct permissions and you have the standard files (*Makefile,* *p01.cc*, *p01.h*, *p01.o*, *p01.exe*) in your */p01* directory. There will be a zero tolerance for deviations from this on this quiz.

Copy the code from my quiz directory to your /p01 directory:

ece-000\_[1]: p

/data/courses/ece\_1111/2025\_01\_fall/quizzes/qu\_06/picone\_joseph/p01

ece-000\_[1]: d

total 16

drwxr-xr-x. 2 picone ece\_1111 66 Oct 9 22:59 ./

drwx------. 3 picone ece\_1111 17 Oct 9 22:46 ../

-rw-r--r--. 1 picone ece\_1111 11 Oct 9 22:59 bytes.dat

-rw-r--r--. 1 picone ece\_1111 427 Oct 9 22:59 Makefile

-rw-r--r--. 1 picone ece\_1111 463 Oct 9 22:59 p01.cc

-rw-r--r--. 1 picone ece\_1111 298 Oct 9 22:59 p01.h

Compile and run the program this way:

ece-000\_[1]: sd p01

/data/courses/ece\_1111/2025\_01\_fall/quizzes/qu\_06/picone\_joseph/p01

ece-000\_[1]: make

g++ -g -c p01.cc -o p01.o

g++ -g -lm p01.o -o p01.exe

ece-000\_[1]: p01.exe bytes.dat

number of bytes read = 11

ece-000\_[1]:

In this quiz, you will add code to my program that converts the bytes in the array x[] to a combination of atomic types. There are 11 bytes in the array. This array could consist of 11 one-byte characters. Or it could consist of 10 two-byte integers plus a one-byte character. You will have to figure out exactly what is held in these 11 bytes. They could be any combination of atomic types.

How will you know when you decode it properly? If you decode it properly, all the values will contain the number “-27”. The floats or doubles will be equal to “-27.00”.

Your program should print to *stdout* the values you decoded and the types of those variables. I’ll leave the details to you, but from your output it should be clear what the types and values are.

If the solution doesn’t compile, you will receive a maximum score of 50. To score above 50, you must decode one or more of the data types correctly.

This type of exercise is very close to what hackers do when they break into a system or decode corrupted data on a disk. If you think about this long enough, you will realize there are only a finite combination of values that can add up to a length of 11 bytes. Hint: try decoding the last byte first.