E0 251: Programming Assignment 2
Due before 11.59 PM on 10-02-2022

Large Integer Arithmetic

You will represent an arbitrarily large unsigned integer I by a singly linked list of its DIGITs d_1 , d_2 , ... , d_m , where d_1 is the least significant DIGIT of I and d_m is the most significant DIGIT of I. Assume that the DIGITs are from a number system with base $1{,}000_{10}$. Implement each DIGIT as a C unsigned int with DIGIT values in [0, 999].

a) Write C functions to perform the arithmetic operations addition and multiplication of large unsigned integers.

b) Write a C function to input a large unsigned integer from stdin as comma separated DIGITs in decimal, ordered on decreasing significance and terminated by a $

c) Write a C function to output a large unsigned integer to stdout in the format used for input, showing each DIGIT as 3 decimal digits.

d) Write a C main() which will repeatedly accept (from stdin) and evaluate, large integer infix expressions on large unsigned integer constants using the operators "+" (addition) and "*" (multiplication). An expression is terminated by "=". On encountering an "=", the value of the expression is to be printed to stdout. Evaluate expressions from left to right. The main() loop should terminate when it receives an empty expression (terminated by =).

For example, the 2 line input
111,041,411,111,011$ + 222,222$ * 003$ =
=
results in the output
333,124,233,999,699$

POSSIBLE EXTENSION (no extra credit): Instead of large unsigned integers, provide the specified functionality for large *signed* integers by modification of your code to use the 1,000's complement representation.