## Homework Assignment #5 Due: October 20, 2016 at 4:00 p.m.

1.

Consider the alphabet  $\left\{ \begin{pmatrix} 0\\0\\0 \end{pmatrix}, \begin{pmatrix} 0\\1\\1 \end{pmatrix}, \begin{pmatrix} 0\\1\\0 \end{pmatrix}, \begin{pmatrix} 0\\1\\1 \end{pmatrix}, \begin{pmatrix} 1\\0\\1 \end{pmatrix}, \begin{pmatrix} 1\\0\\0 \end{pmatrix}, \begin{pmatrix} 1\\0\\0\\1 \end{pmatrix}, \begin{pmatrix} 1\\1\\0 \end{pmatrix}, \begin{pmatrix} 1\\1\\1\\0 \end{pmatrix}, \begin{pmatrix} 1\\1\\1\\1 \end{pmatrix} \right\}$ . We shall use strings in this alphabet to describe three integers: one for the top row of bits, one for the middle row and one for the bottom row. Each integer is represented in binary. For example, to represent the three integers 5, 7 and 35 (whose binary representations are 101, 111 and 100011, respectively), we use the string  $\begin{pmatrix} 0\\0\\1 \end{pmatrix} \begin{pmatrix} 0\\0\\0 \end{pmatrix} \begin{pmatrix} 0\\0\\0 \end{pmatrix} \begin{pmatrix} 1\\1\\0 \end{pmatrix} \begin{pmatrix} 0\\1\\1 \end{pmatrix} \begin{pmatrix} 1\\1\\1 \end{pmatrix}$ . (Note that we add 0's at the left end of the binary representations of the numbers in order to pad them so that all three rows are of the same length.) Let *MULT* be the language of all strings that represent correct binary multiplications, i.e, where the number represented by the third row is the product of the numbers represented

in the first two rows. For example, the string  $\begin{pmatrix} 0\\0\\1 \end{pmatrix} \begin{pmatrix} 0\\0\\0 \end{pmatrix} \begin{pmatrix} 0\\0\\0 \end{pmatrix} \begin{pmatrix} 1\\1\\0 \end{pmatrix} \begin{pmatrix} 0\\1\\1 \end{pmatrix} \begin{pmatrix} 1\\1\\1 \end{pmatrix}$  is in

MULT, since  $5 \times 7 = 35$ .

Prove that the language *MULT* is not regular.

(Recall that we proved in class that the analogous language for addition is regular, so multiplication is harder than addition.)

2. Read the definition of minimum pumping length in problem 1.55 on page 91 of the textbook. What is the minimum pumping length for the language represented by the regular expression 1\*001\*0?