Due 3/15

Construct a PDA that accepts $\{ x\#y\#z \mid x, y, z$ in $\{0, 1\}^+$ with $x \approx y$, or $x \approx z$, or $y \approx z \}$.

Define $x \approx y$ as follows: Let $x = x_1x_2 \ldots x_n$ and $y = y_1y_2 \ldots y_m$, and let n' and m' be the largest odd values less than or equal to n and m respectively. Let $x' = x_1x_3 \ldots x_{n'}$ and $y' = y_1y_3 \ldots y_{m'}$. Then $x \approx y$ if $x' \neq y'$ (that is, $x_1x_3 \ldots x_{n'} \neq y_1y_3 \ldots y_{m'}$).

For your PDA to work correctly it will need to be non-deterministic. You can assume that you will always be given a valid string – that is, the input will always contain two #s, and x, y, and z will be strings over {0, 1} of length greater than 0.

My PDA accepts x#y#z under any of the following conditions: $|x'| \neq |y'|$ or $|x'| \neq |z'|$ or $|y'| \neq |z'|$ or there exists odd value i with $x_i \neq y_i$ or $x_i \neq z_i$ or $y_i \neq z_i$.

You should use the solution for problem 2.22 as the basis for the solution to this problem (we went over problem 2.22 in class on 2/26/2019). Problem 2.22 is to show that $\{ x\#y \mid x, y$ in $\{0, 1\}^*$ with $x \neq y \}$ is context free. Based on the solution to problem 2.22, constructing the PDA for this programming assignment should be relatively straight forward.

Your PDA will need to be able to handle input strings of length up to about 200 symbols using at most 1GB of memory (the memory should not be an issue, but you want to stay away from using the transition $\varepsilon, \varepsilon \rightarrow \varepsilon$ in too many places).

My simplistic PDA has 21 states, and my more complicated version has 15 states, and both are able to process the test strings with the default amount of memory allocated by the Java runtime machine (64MB).

If you are having memory issues, try starting JFLAP from the command line allocating more memory. As an example, from the directory that JFLAP is in, enter "java -Xmx500m -classpath JFLAP.jar JFLAP". The -Xmx500m tells java to allocate 500MB of memory to the heap. By default, java only allocates 64MB to the heap.

E-mail the JFLAP file to me (david.garrison@binghamton.edu) by 11:59:59.999pm on the date due. The filename is to be your last name in lower case followed by either "_p3.jff" (as an example, my filename would be "garrison_p3.jff"). The subject of your e-mail is to be "CS 373 program 3".

For this programming assignment, and the remainder of them for the semester, you need to follow my submission directions – filename (lower case last name followed by "_p3.jff") and e-mail subject ("CS 373 program 3").

You may use JFLAP version 8 if you have problems with epsilon transitions with version 6.4.