Ling 5801: Problem Set 1

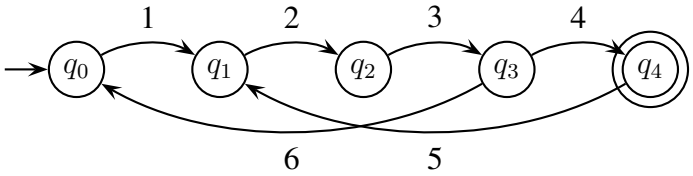
Due via Carmen dropbox at 11:59 PM 9/5.

1. [5 pts.] Draw an FSA that recognizes the following language: (0*1|1*0). (The bar outscopes the other operators, so this is equivalent to: (0*1)|(1*0).) Try to define your FSA using the fewest states possible.

2. [5 pts.] Write a regular expression that recognizes any string of 1's and/or 0's containing an odd number of 0's. Try to make your regular expression as concise as possible.

3. [5 pts.] Draw an FSA that recognizes any string of 1's and/or 0's containing an even number of 0's and an odd number of 1's. Try to define your FSA using the fewest states possible.

4. [1 pt. extra credit – tricky!] What language does the following FSA generate? Answer using a regular expression. Try to make your regular expression as concise as possible.



5. [30 pts.] PROGRAMMING:

   (In general for your programming problems you should hand in the following as separate files (NOT a single zip-file):

   - a copy of your Makefile (NOTE: you may have to rename it 'Makefile.txt' to upload)
   - a copy of each script you write,
   - a representative sample of each source (input) file you use,
   - a representative sample of each target (output) you produce.

   You may use the sample Makefile on the course home page to obtain .txt files from copyright-friendly wikipedia articles. e.g. Speed_of_light has several big numbers in it.)

   Write a Makefile that can do all of the following —

   (a) [10 pts.] Make a target '%.numlines' file, consisting of the *lines* in a corresponding source '%.txt' file that contain a number, defined here to be a maximal sequence of digits, commas, and decimal points ending in a digit. If a number in prose is immediately followed by a comma or decimal point, you should not include the comma or decimal point in the number.

   (b) [10 pts.] Make a target '%.nonums' file, by replacing every number in a corresponding source '%.txt' file (numbers as defined in the previous sub-problem) with the string 'NUM'. (Remember that numbers may include commas and decimal points.)

(c) [10 pts.] Make a target '%.numclass' file, by replacing every number in a corresponding source '%.txt' file (numbers as defined in the previous sub-problem) with the string 'BIGNUM' if the number is greater than or equal to 1,000, or the string 'SMALLNUM' if the number is less than 1,000. (Remember that numbers may include commas and decimal points.)