This homework covers the reading from Chapter 3, Sections 3 and 4 and from this weeks reading on the web. It is due by the end of Wednesday, March 24, and will be accepted late with a 10% penalty until noon on Sunday, March 28.

1. (*9 points*) For each of the following languages, draw a transition diagram for a DFA that accepts the language. That is, it accepts all the strings in the language and no other strings. (The alphabet for the DFA is the same as the alphabet for the language.)

a) $\{w \in \{a,b,c\}^* \mid$ the number of a's plus the number of b's in w is **not** a multiple of 3 $\}$ [Note that the alphabet also includes c!]

b) $\{x \in \{a,b,c\}^* \mid x$ contains a c and there are no a's before the first $c\}$

c) $\{y \in \{a,b\}^* \mid y$ ends with the string ababb$\}$

2. (*5 points*) Suppose that a DFA M is defined as $M = (Q, \Sigma, p_1, \delta, F)$, where:

$Q = \{p_1, p_2, p_3, p_4\}$ $\quad \Sigma = \{a, b, c\}$ $\quad F = \{p_2, p_4\}$

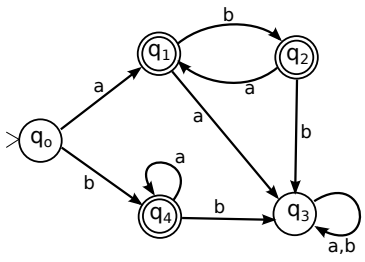
and δ is given by the table shown at the right.

	p_1	p_2	p_3	p_4
a	p_2	p_2	p_3	p_3
b	p_4	p_3	p_3	p_4
c	p_1	p_3	p_3	p_4

a) Draw a transition diagram for M.

b) Based on your diagram, write a regular expression for the language that is accepted by M, and briefly explain your reasoning.

3. (*5 points*) Consider the DFA that is defined by the transition diagram shown at the right.



a) Suppose that this DFA is given formally as $M = \{Q, \Sigma, q_o, \delta, F\}$. Identify Q, Σ, δ, and F. For δ, give the transition table.

b) Find a regular expression for the language that is accepted by this DFA. Explain your reasoning.

The remaining exercises are based on a file, *data.txt*, which you can get from the directory /classes/cs229 on math.hws.edu or through the following link

http://math.hws.edu/eck/cs229/s21/data.txt

This is based on an anonymized file of grade data from the registrar, but the grades have been randomized to make the data totally meaningless. I urge you to actually do the exercises on a computer, but you are only asked to report how the exercises can be done. The first three exercises could be done in a text editor that supports regular expressions. You could, for example, add the file to a project in Eclipse and use Eclipse's search-and-replace feature. You could do the same exercises on the command line in Linux or, I believe, on a Mac, with a command that uses *perl -pe*. The last two exercises require the command line.

4. (*2 points*) Dates in the file are given in the format Month/Day/Year. You would like them to be in the form Day-Month-Year. For example, you would transform 3/17/2021 to 17-3-2021. How could you use regular expression search-and-replace to make the change? What search expression and replacement text would you use?

5. (*2 points*) The first item on each line of the file is a four-digit code for the academic term in which the course was taken. The first digit is always 1, the second and third digit give the last two digits of the year, and the fourth digit is 2 for Spring term, 4 for Summer term, or 6 for Fall term. All of the terms in the file are in the 21st century. You would like to have the terms listed in a more standard format such as Spring 2009, Summer 2018, and Fall 2015. What **three** regular expression search-and-replace operations could you use to make the change?

6. (*2 points*) Each line of the file contains five fields, separated by commas. You would like to discard the last two fields, interchange the first two fields, and add spaces after the commas. So, for example, the line

1152,MATH 110,B-,23,5/29/2013

would be transformed to

MATH 110, 1152, B-

This change can be made using regular expression search-and-replace. You want to be careful to match the whole line and pull out the parts that you need. What search expression and replacement text could you use? Hint: .* can be useful here.

7. (*2 points*) The file includes lines with grades such as CR and W. You only want the data for letter grades. How could you use the *egrep* command to retain only the lines that contain letter grades? You need to find a regular expression that matches only those lines? A letter grade consists of one of the letters A, B, C, D, or F, optionally followed by a + or - sign. Some of the letters can occur in the file in other places besides letter grades, so be careful. Hint: Make use of certain commas in the file.

8. (*2 points*) This problem does not use regular expressions. The fourth field of the file contains grades of various kinds such as A+, NC, and VW. You would like to know, without carefully reading the file, exactly what grades are possible. What command could be used on the command line to get a list of the grades that occur, without any repetition in the list? This can be done using a combination of the *cut* and *sort* commands. What would you add to the command to easily find out how many different possible grades there are?