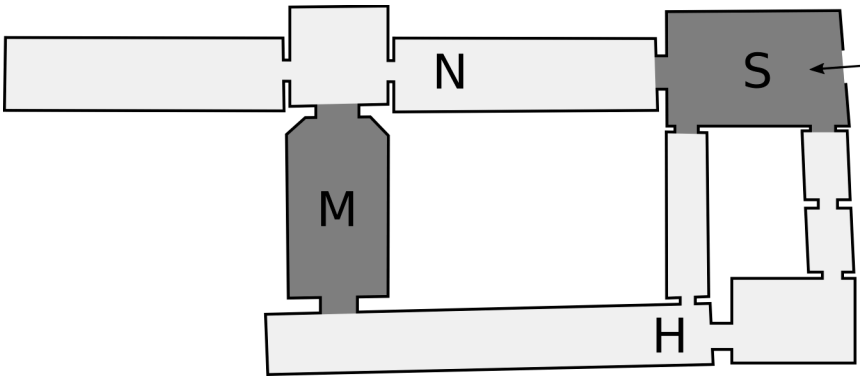
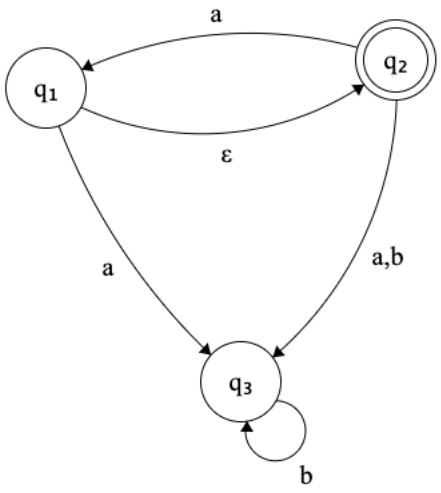
CSC333 (002) Jennings
Homework 2, Fall 2019
There are 10 questions. Please answer all of them, noting that the last part of the last question is optional.

1. Name: _____



2. The Floor plan above depicts the Denon Wing of the Louvre in Paris. In this (simplified) schematic, we have marked four areas, labeled S, H, M, N. Area M contains da Vinci's *Mona Lisa* and area N contains David's *The Coronation of Napoleon*.
   A tour through this part of the Museum is a string over {S,H,M,N} which represents a walk through these areas. A tour needs to start and end in the area S and needs to follow the floor plan.
   For example, SMS is not a tour since we cannot visit M from S without going through either N or H. The string SNMHHS is not a tour since we cannot visit H twice without going somewhere else first.
   The string SNMHMHS is an example of a tour.
   Design (draw) a DFA which recognizes the language of all tours that contain either the *Mona Lisa* or *The Coronation of Napoleon* or both.

3. The diagram below is an NFA. Convert it to a DFA using the technique shown in class (which is Theorem 1.39 in the textbook).



4. Convert each of the following regular expressions to an equivalent NFA.
    a. ϕ (the empty set, i.e. the language containing no strings at all)

    b. ε (the language containing only the empty string)

    c. $(0 \cup 1)^*000(0 \cup 1)^*$

    d. $(((00)^*(11)) \cup 01)^*$

5. Every NFA is also a DFA: (circle one) **true** or **false**

6. Every NFA can be converted to an equivalent DFA: (circle one) **true** or **false**

7. A DFA must have exactly s arrows leading out of every state, where s = _____.

8. If you convert an NFA into a DFA, the number of states in the DFA will be **more** or **less** (circle one) than the number of states of the NFA.

9. In the worst case, the NFA to DFA conversion can produce a DFA with how many states? Assume the NFA has k states. *(Hint: You do not need big-Oh notation here.)*

   Answer: _____

10. For this problem, use the program `egrep`, which you installed as part of the previous homework. The data files you need for input are available on Piazza under Resources, in the Homework section. Use `egrep` at the command line (a terminal window), and cut-and-paste the command and its output here, as the answer to these questions. Part (a) is done for you.

    a. Use `egrep` to find all of lines in `sh.txt` that contain the fixed string "candle".

```
~/Classes/333/jamie/homework$ egrep 'candle' sh.txt
guttering candle in the other. Anyhow, he never got tallow-stains from
beside him. By it he laid the box of matches and the stump of a candle.
a candle in her right. It was the same good friend whose warning I had
~/Classes/333/jamie/homework$
```

    Notes:
    - *My shell prompt is shown next to the command I typed (first line) and again at the end (making sure that I copied the entire command output).*
    - *Please use a font that is 10pt or larger and mono-spaced, to make it easier for grading.*
    - *I enclosed the regular expression in single quotes because otherwise the shell may do some substitutions (for characters like *), and we don't want that.*

    b. The name of each work in this collection has a Roman numeral in front of it. Using `egrep`, write a regular expression to match any Roman numeral from I through XII, followed by a period, followed by a space. (You will need to escape the period using a backslash, i.e. write it as `'\.'`)

    You should see between 20--40 lines of output, which you do NOT need to paste here. How many lines did your regular expression match?

    Answer: _____

    Note: If you are on a Unix/Linux operating system or you have Unix tools installed, you can pipe the output of `egrep` into the `wc` command to get this answer without counting the lines yourself.

c. Change the regular expression you wrote for (b) by putting two spaces after the period, instead of the one used in (b). You should now get only the table of contents of this collection. Show your command and output:

d. If you examine the output from step (b) above, you'll see that each work in the collection starts with its title, with a Roman numeral in front of it. (You can verify this by browsing the file in a text editor or browser.) Modify the regular expression from (b) to find all those title lines. Make sure you get all 12 titles and only those as output!
   Paste your command and output here.

e. Use the same command as (d) but add the option ‘-n’ to the egrep command, so that it will print the line numbers for each line of output.
Paste your command and output here.

f. Find all the questions containing the word ‘body’. Like any English sentence, a question starts with a capitalized word, and ends in a question mark.
Paste your command and output here.

g. Find all exclamations containing the word ‘body’. Like any English sentence, an exclamation starts with a capitalized word, and ends in an exclamation point.
Paste your command and output here.

h. Using the command from (g), add the `-o` option to `egrep` so that it prints only the part of the line that matched the regular expression.
Paste your command and output here.

i. Bonus question (extra credit): Devise a regular expression that matches any Roman numeral, or argue that it cannot be done.