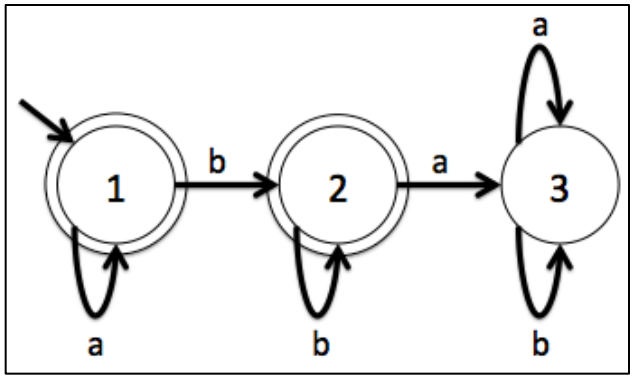
Homework #5: Finite Automaton (100 pts)

In this assignment, your job is to implement a flexible **discrete finite automaton** in C++. In particular, your solution must meet the following requirements:

- Takes **one** cmd line argument that identifies a *finite automaton specification file*
  - A *finite automaton specification file* will have the following format:
    - 1st line: *1 or more* integers identifying **Q**
    - 2nd line: *1 or more* characters specifying **Σ**
    - 3rd line: *1* integer identifying $\mathbf{q_0}$
    - 4th line: *1 or more* integers specifying **T**
    - 5th line until EOF: 1 integer, 1 character, and 1 integer identifying **δ**

      - Example file for a finite automaton that accepts regular language L = *a*b**

```
1 2 3
a b
1
1 2
1 a 1
1 b 2
2 a 3
2 b 2
3 a 3
3 b 3
```



- After reading the machine specification file, your solution must output the finite automaton's structure. At a minimum, your solution must identify:
  - $\mathbf{q_0}$
  - **T**
  - **δ**

- Takes *1 or more* lines of user input on **stdin** (until EOF)
  - Each line of user input will contain a sequence of symbols

- Your solution must run **each** *sequence* of symbols through the **finite automaton** defined by the *machine specification file*

- For *each* sequence, your solution must output either "**accept**" or "**reject**"

- For *each* sequence, your solution must output the **path** through the finite automaton

EXAMPLE

```
UNIX> cat "a*b*.txt"
1 2 3
a b
1
1 2
1 a 1
1 b 2
2 a 3
2 b 2
3 a 3
3 b 3

UNIX> ./a.out "a*b*.txt"
q0 = 1*
1,a -> 1*
1,b -> 2*
2,a -> 3
2,b -> 2*
3,a -> 3
3,b -> 3
Enter a string to check
<ENTER>
accepted
1*
==================
Enter a string:
a
accepted
1* -> 1*
==================
Enter a string:
b
accepted
1* -> 2*
==================
Enter a string:
aa
accepted
1* -> 1* -> 1*
==================
Enter a string:
ab
accepted
1* -> 1* -> 2*
==================
Enter a string:
ba
rejected
1* -> 2* -> 3
==================
Enter a string:
<CNTL-D>

UNIX>
```

```
UNIX> cat "(a+b)*bb(a+b)*.txt"
1 2 3
a b
1
3
1 a 1
1 b 2
2 a 1
2 b 3
3 a 3
3 b 3

UNIX> ./a.out "(a+b)*bb(a+b)*.txt"
q0 = 1
1,a -> 1
1,b -> 2
2,a -> 1
2,b -> 3*
3,a -> 3*
3,b -> 3*
Enter a string to check
aaababa
rejected
1  -> 1  -> 1  -> 1  -> 2  -> 1  -> 2  -> 1
==================
Enter a string:
aaabbaa
accepted
1  -> 1  -> 1  -> 1  -> 2  -> 3* -> 3* -> 3*
==================
Enter a string:
<CNTL-D>

UNIX>
```

HINTS

- start early
- work incrementally
- plan your solution before you code
- **test, test, test**

- ifstream
- getline( )
- vector
- map
- map *
- struct State
- class DFA (will make your code cleaner)
- strtok( )
- sscanf
- printf
- ternary operator

- ***You can assume the machine spec file will be correct!! (No need to error check)***