# State Machine to Validate Input

**Overview:** Your assignment is to construct a program that validates input using a state machine.

**Configuration Rules:** Your program must be called “validator” and take 2 command line parameters – a state machine filename, then a filename of inputs to validate. For example:

validator myStateMachine.txt myInputs.txt

**State Machine:** The state machine will come from a text file in the following space delimited format:

stateNumber inputCharacter transitionsToState

For example:

0 A 1

0 a 1

1 B 2

1 b 2

2 C 999

There are three terminal situations:

999 – success – when this state is reached, print “Success” and end.

failure (no transitions match) – print “Failure at position \_\_\_, found character \_.”

failure (input string ends early) – print “Input string ended before success transition.”

Note – each state can have ANY NUMBER (1 or more) of transitions to other states. The transitions do not have to happen in any specific order (for example – state 1 could transition forwards to state 4 and state 5 could transition backward to state 2). States will be sorted in the text file (0,1,2,3,4, etc.).

**Input File:** The inputs to validate is some number (1 or more) of lines of text. Example:

abc

AbC

**Restrictions:** You may use Java file I/O classes and collections. You may not use any state machine classes that are included with Java or that you have found elsewhere. **YOU MUST WRITE ALL OF THE CODE YOURSELF except for the Java I/O classes and collections.**

**Testing:** Please be sure to test your code very thoroughly. It will be graded with state machines and text files that you do not have access to. **I encourage you to SHARE your test files with each other. You may NOT share your code, though.**