Name: Jaiswal, Devesh Date: April 3rd,2024

NYU ID: N14682845 dj1380

Course Section: Data Comm & Networks CSCI-GA 2262 Section 001

**Lab 6**

Total in Points (100 points in total):

Professors Comments:

Affirmation of my Independent Effort: Devesh Jaiswal

ReadMe:

My program source code contains 6 classes. The main program to run is in the class named FiniteStateMachine.java. I wasn’t sure what the start state should be so in line 67: FSM fsm = new FSM("fsm", established);. I set the start state as established but this can be modified as necessary.

So when first entering the program the start state will be established and then can take in transitions that can occur at established firstly since currently that is my hardcoded start state since I wasn’t sure what the start state should be.

Furthermore, I wrote this in Eclipse so my classes(.java files) are all within a src folder of a project named DCN\_LAB6, and in the Referenced Libraries classpath, I have added the .class files of the Fsm package given as a jar file I created. So the .class files are within the Referenced Libraries→Fsm 2.jar→ Fsm Package. With this layout the program works correctly the only thing is that as stated above I have hardcoded the start state which can be modified since I wasn’t sure which state the start state is. It takes in input until the end of the file/input is reached which in terminal or standard input is control + d.

When extending/inheriting the .class files from the Fsm package we prepend it with Fsm. since the .class files are within the Fsm package. For the classes where we have inherited from the Fsm package given we essentially use the inheritance property to implement all the methods. The only which was different was the action class where we output the appropriate message according to the event given. Comments are in my main file(FiniteStateMachine.java) about how I implemented the logic for the FSM.

Regardless, we run from FiniteStateMachine.java and the program should work as expected. I have submitted a jar file of my own which only contains the .java files of my program so when running must include the jar file of the .class files of the Fsm Package given which can be included in the referenced libraries classpath as mentioned if testing in Eclipse. So essentially to run we can use the jar file but must also include the jar file of the FSM package. I have also included the jar file for the Fsm Package given from the assignment just for ease of use even though it is not required. Below are instructions to test through the terminal.

**STEPS TO RUN ON THE TERMINAL:**

-I am submitting a jar file named dj1380.jar with my .java files. Also submitting an Fsm2.jar file which is the jar file for the .class files of the Fsm Package given in the assignment even though it is not required just for ease of use. If not using that you can also just use your own jar file of the .class files given in the assignment. Just make sure the appropriate name is then used in steps 4 and 5 instead of Fsm2.jar if using your own created jar file of the .class files. But this program guaranteed works if using Fsm2.jar.

So you can:

**1 -Open the zip file I submitted.**

**2- Go to the directory of the folder containing the files from the opened zip file in the terminal. Probably named Devesh\_Jaiswal\_Lab\_6→cd Devesh\_Jaiswal\_Lab\_6**

**3- jar xf dj1380.jar**

**4- javac -cp "Fsm2.jar" \*.java**

**5- java -cp ".:Fsm2.jar" FiniteStateMachine**

since FiniteStateMachine is the name of my main program.

The program exits when the end of file/input is reached or essentially when **cntrl + d** is entered into the terminal which signals the end of file/input.

The document in that zip file is this document which i used to serve as a report and a readme.