TCS Spring ‘17

Assignment 1

FSA Simulator

Due: Feb, 8 ‘17

Problem Definition

You are asked to develop a program which simulates a Finite State Automata. The simulator will take a description of FSA in a text file, build an internal model and run a number of strings through the produced model. For each string, the simulator should output whether a string belongs to the respective language represented by the automaton or not as well as produce a trace of execution. You may assume that the input is always valid and it is always possible to build an automaton with a given description.

Input

The first line of the input file will consist of an integer number *k* specifying the number of test cases. This is followed by the data specifying each test case. There will be no blank lines in the input file.

Each test case consists of two parts - the automaton description and a set of test strings.

The automaton description consists of exactly 5 lines:
1. Set of states
2. Input alphabet
3. Initial state
4. Set of final states
5. Set of transition in the form of “current_state(input_symbol)->new_state”

The automaton description is followed by the number *n* specifying the number of test strings for the automaton. The following n lines each contain a single test string.

Output

The first line of your output file should contain your name.

Then, for each test case in the input file, output the following:
1. Test case number
2. For each test string in a respective test case, output whether it belongs to the language defined by the automaton (True or False) and print the execution trace in the form of “inital_state->state_1->state_2…” (see Sample Output for details).
   NOTE: The output for each test string should be on a separate line!

Sample Input

```
2
{q0,q1,q2}
{a,b}
q0
{q2}
{q0(a)->q1,q0(b)->q0,q1(a)->q2,q1(b)->q0,q2(a)->q2,q2(b)->q0}
2
abaa
bab
{q0,q1,q2,q3}
{a,b,c}
q0
{q3}
{q0(a)->q1,q0(b)->q0,q0(c)->q0,q1(a)->q2,q1(b)->q0,q1(c)->q0,q2(a)-
>q2,q2(b)->q3,q2(c)->q0,q3(a)->q3,q3(b)->q3,q3(c)->q3} #On the same line
4
abbbaaa
aabaababb

bbbbb
```

Sample Output

```
JohnDoe
1
True,q0->q1->q0->q1->q2
False,q0->q0->q1->q0
2
False,q0->q1->q0->q0->q0->q1->q2->q2
True,q0->q1->q2->q3->q3->q3->q3->q3->q3->q3
False,q0
False,q0->q0->q0->q0->q0->q0
```