CS335 (Compiler Design)

Programming Assignment

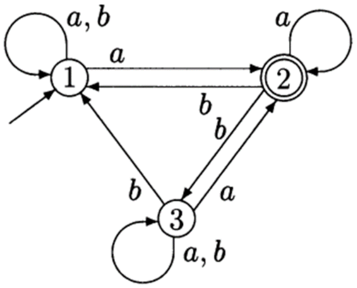
Write a program to convert an NFA with/without epsilon transition to its corresponding DFA. Your application should first ask for the following input:

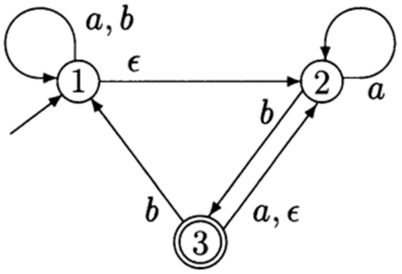
- No. of states in NFA and their values
- The start state and final state(s)
- The input symbols
- The transition, i.e., move(state, input symbol) for all possible cases (including epsilon transitions)

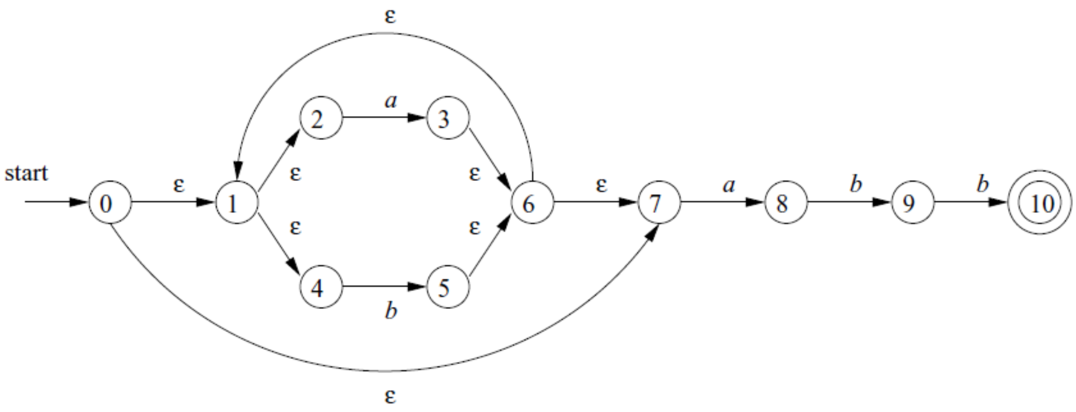
Following is the NFA to DFA conversion algorithm:

```
1. Initially, the list of states in M' is empty.
2. Create the starting state named after the ϵ-closure of M's starting
   state.
3. While (there is an uncompleted row in the table for M') do:
      a. Let x = [s1, s2, s3, ..., sk] be the state for this row.
           i.  For each input symbol a do:
          ii.  Find the ϵ-closure of N({s1,..., sk}, a) = some set we'll
               call T.
         iii.  Create the M'-state y = [T] corresponding to T.
          iv.  If y is not already in the list of M'-states, add it to
               the list.
           v.  Add the rule N'(x, a) = y to the list of transition rules
               for M'.
4. Identify the accepting states in M'.
```

The output of your implementation should be the transition table of the resulting DFA in a proper format. You need to demonstrate the working of the algorithm on the following three NFAs.


a, b
a
a
1
2
b
b
b
a
3
a, b


a, b
ϵ
1
2
a
b
b
a, ϵ
3


ε
start
0
ε
1
ε
ε
2
a
3
4
b
5
ε
ε
6
ε
7
a
8
b
9
b
10
ε