IT 328, Introduction to the Theory of Computation

Programming Assignment 3 : Minimize DFA

Due date: March 30, 2017, Thursday, 50 points (70% on programs, 30% on report)

Based on the program for previous assignment where an NFA is converted to an equivalent DFA, you are asked to minimized the DFA. Then use the minimized DFA to parse strings in the same text file, `inputStrings.txt`. The DFA's you have to minimize are the results of your previous assignment converted from the same NFA's, i.e., `nfa1`, `nfa2`, `nfa3`, `nfa4`, and `nfa5`.

You can copy and modify your own programs for the previous assignment and name the program as `parse.java`. I will compile and run your program as follows:

```
java parse nfa2.nfa inputStrings.txt
```

where the arguments are in the same format as described in the previous assignment. All five NFA's files will be tried. Your program should convert the input NFA to an equivalent DFA (this is the task of the previous assignment), then minimize the DFA into an optimal DFA. Finally, use the optimal DFA to parse every string in the input string file. Your program should print out the following:

(1) The minimized DFA (after name each state with a unique number)

(2) The results of parsing all strings in `inputStrings.txt`.

Here is the sample output of `java parse nfa2 inputStrings.txt`

```
Minimized DFA from nfa2:
 Sigma:     a     b
 ------------------
     0:     1     0
     1:     1     0
 ------------------
0:  Initial State
1:  Accepting State(s)
The following strings are accepted:
aabaa
aaaaa
....
```

Final Step: As before, select a secret name, say "`peekapoo`", and

```
bash /home/ADILSTU/ccli/Public/IT328/copy328.sh peekapoo
```

Note that your original works have to be in the same directory `~/IT328/nfa/` used in your previous assignment.

You have to follow the submission guidelines, i.e., cover page (that contains assignment number, student's names, student **ULID**, and secret directory), summary, source code(optional), output, folder, and so on. **No late work will be accepted.**