



Compiler Design spring 1402

Sharif University of Technology

Assignment 1

Due: Wednesday 25/12/1401 at 11:59pm

Released: Friday 12/12/1401

This assignment covers lectures 1 to 3 about regular languages, finite automata and lexical analysis.

10 % of the points will be applied if you upload 4 days earlier than the deadline

5 % of the points will be applied if you upload 2 days earlier than the deadline

you are only allowed to use notation provided on second

page of third slide , to write regex.

1. Provide a regular expression for the following languages. (Suppose all languages are defined on the alphabet $\Sigma = \{0,1\}$) (30 points)

- A. All strings containing an even number of 0's or even number of 1's: (5 points)
- B. All strings containing an even number of 0's and an even number of 1's: (5 points)
- C. All strings that don't contain the substring 001: (5 points)
- D. All strings that don't contain the substring 111. (5 points)
- E. All strings that contain exactly one substring 111. (5 points)
- F. All strings that have no consecutive 0's and no consecutive 1's: (5 points)

2. Draw an NFA for languages: (10 points)

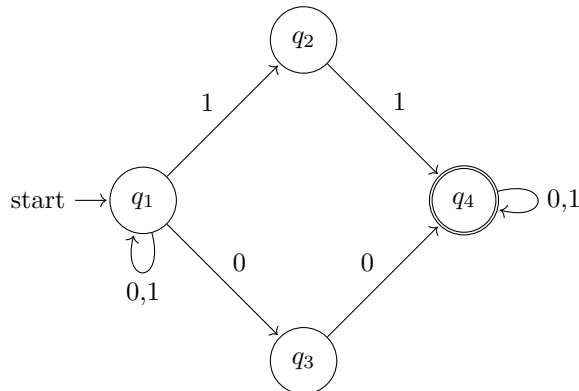
- A. $L_1 = \{ \omega \in \{a,b,c\}^* \mid \omega \text{ is not } \varepsilon \text{ and the last character of } \omega \text{ appears at most twice in } \omega \}$ (5 points)
- B. $L_2 = \{ \omega \in \{0,1\}^* \mid \omega \text{ contains at least two 0s, or exactly two 1s} \}$ (5 points)

3. Draw an NFA for given regexes: (10 points)

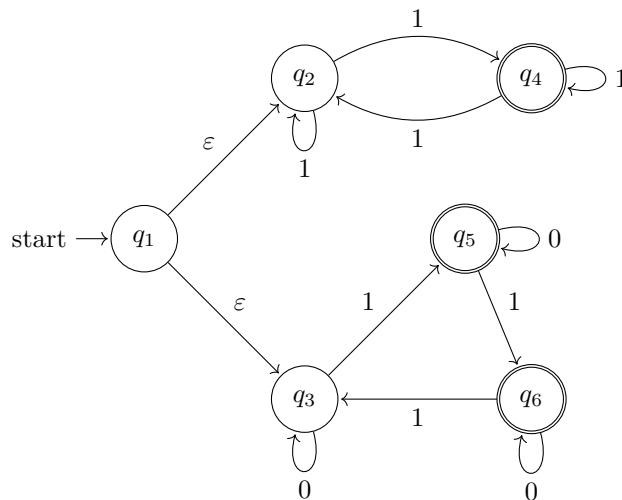
- A. $\varepsilon \mid 0 \mid 1 \mid (0 \mid 1)^* (11 \mid 00 \mid 10)$ (5 points)
- B. $((ab)^* c (d \mid e) (f \mid g))^* h$ (5 points)

4. NFA to DFA (20 points)

- A. $\Sigma = \{0,1\}$ (10 points)



- B. $\Sigma = \{0,1\}$ (10 points)



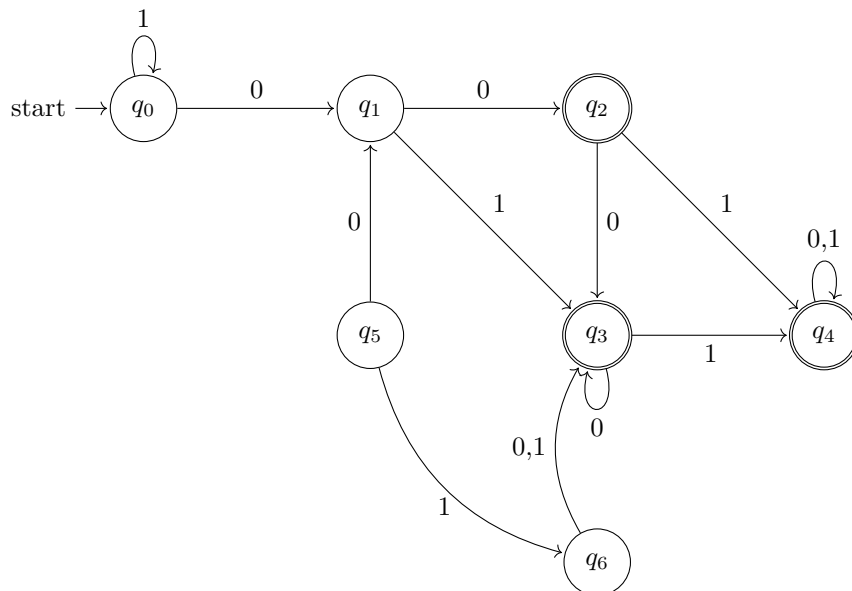
5. Consider the following set of tokens types: (40 points)

Token Type	Description
NUM	Any string matching : $[0 - 9]^+$
ID	Any string matching : $[A - Z a - z][A - Z a - z0 - 9]^*$
KEYWORD	if ,else, void, int, repeat, break, until, return
SYMBOL	$; : , [] () \{ \} + - * = < ==$
COMMENT	Any string between a $/^*$ and a $*/$
WHITESPACE	$\text{blank (ASCII 32), } \backslash n \text{ (ASCII 10), } \backslash r \text{ (ASCII 13), } \backslash t \text{ (ASCII 9), } \backslash v \text{ (ASCII 11), } \backslash f \text{ (ASCII 12)}$

- Draw appropriate DFAs (i.e., similar to the DFAs in pages 56-58 in Lecture note 3) for recognizing these tokens. Note that ID and KEYWORD are recognized by the same DFA, which is almost identical to the one on Page 57 of Lecture note 3!
- In each of these DFAs, specify exactly what characters should be considered as compatible with the 'other' label. Note label 'other' in different DFAs are not necessarily referring to the same set of symbols.
- Then combine these DFAs into a single DFA. The resultant DFA (provided that the DFA is correct and complete) can then be used as a flowchart for implementing a scanner in Programming Assignment 1, to be later released.

6. DFA optimization:

Draw an equivalent DFA for the given DFA with minimum number of states. (10 points)



7. provide a regex for the given DFA: (10 points)

