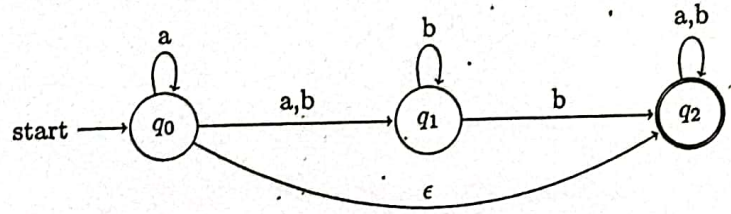


$R = \{ \epsilon, aab, aabb, \dots \}$

10

## Compiler Construction Quiz-2

1. Write regular expression of language having even number of a's and b's. [5 Marks]
2. Create an NFA for regular expression  $b^*(aab(aab)^*)^*$ . Show all the steps. [5 Marks]
3. Convert following NFA to DFA. Clearly show all the steps. [5 Marks]

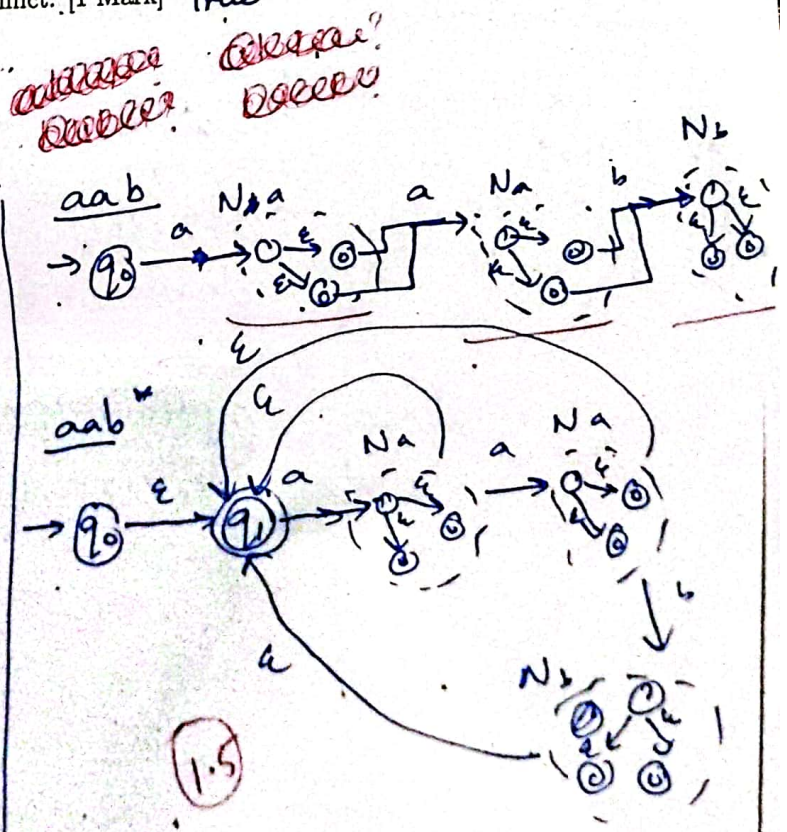
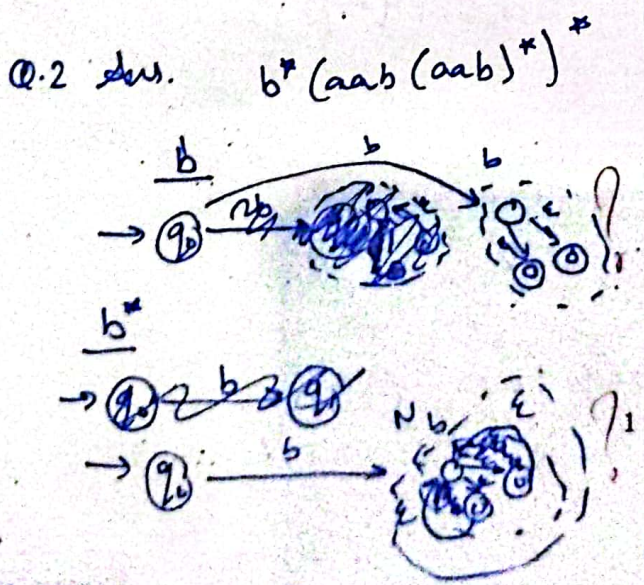


3

### 4. Short-answers

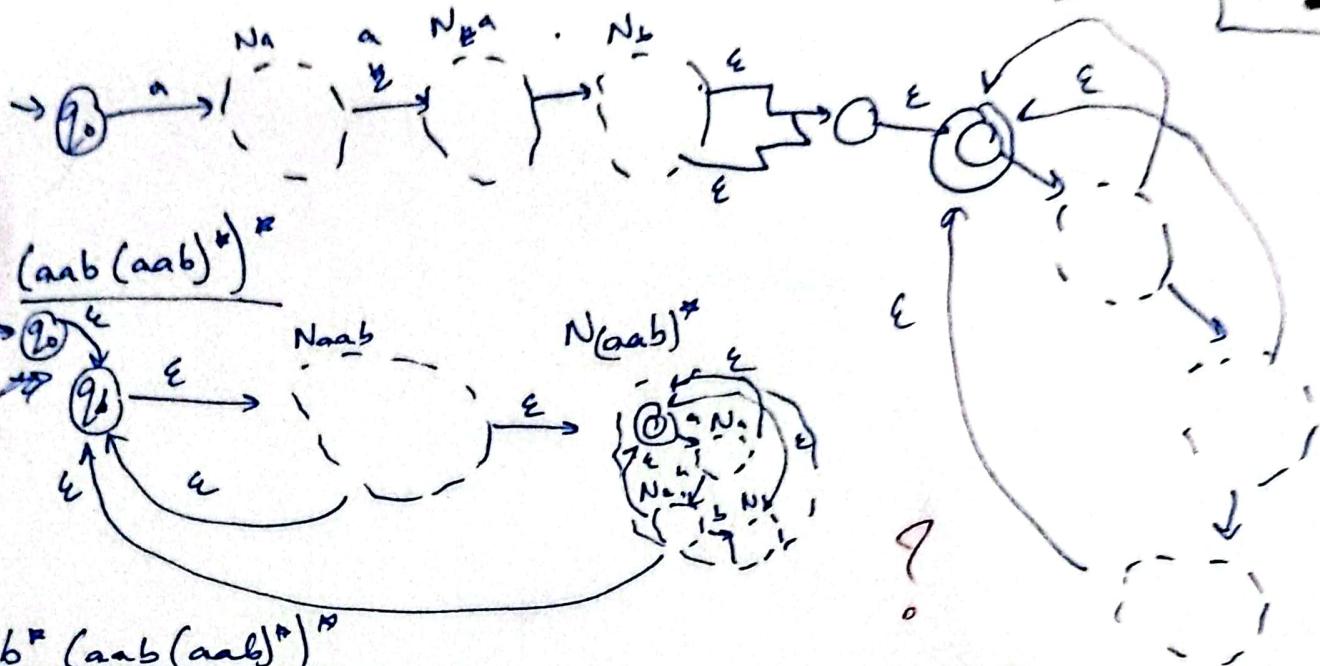
- (a) True/False: Some NFAs more powerful than all DFAs, in terms of the languages they can express [1 Mark] **False**
- (b) True/False: NFAs are faster than DFAs, in general. [1 Mark] **True**
- (c) True/False: NFAs are smaller than DFAs, in general. [1 Mark] **True**
- (d) Although we write regular expressions for lexical analyzer but it is convert for character stream tokenization of source code. [2 Mark]
- (e) True/False: It is common that a string is recognized by multiple regular expressions in lexical analyzer and then we have to add additional criteria to resolve conflict. [1 Mark] **True**

Q.1 Ans. R.E =  $(aa+bb)^*$

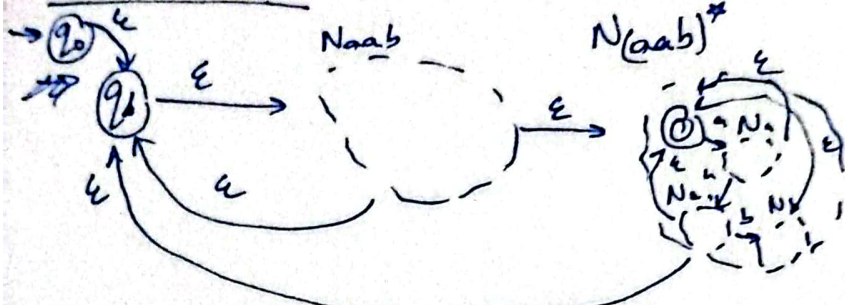




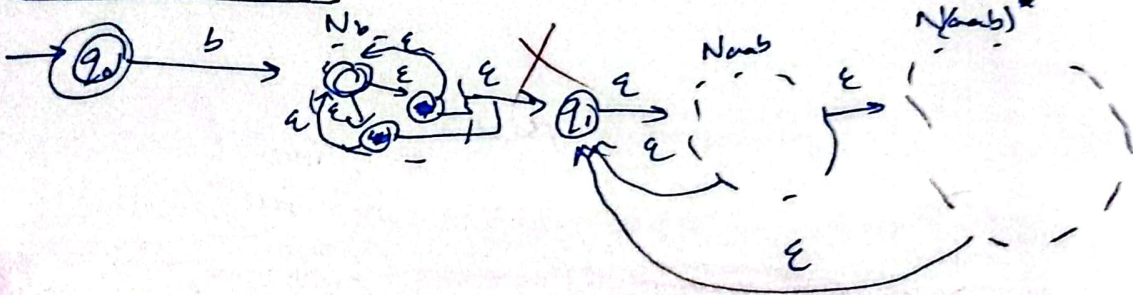
$aab(aab)^*$



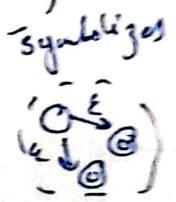
$(aab(aab)^*)^*$



$b^*(aab(aab)^*)^*$



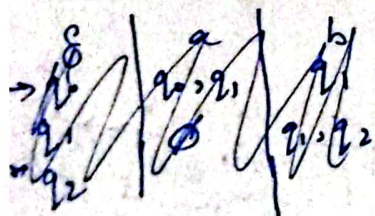
NOTE:



2.3

Transition Table NFA

NFA NFA



$q_0$  closure =  $\{q_0, q_1, q_2\}$   
 $q_1$  closure =  $\{q_1, q_2\}$   
 $q_2$  closure =  $\{q_2\}$

$q_0$  for  $a = \{q_1, q_2\}$   
 $q_0 \xrightarrow{a} q_1$   
 $q_1 \xrightarrow{a} q_2$

