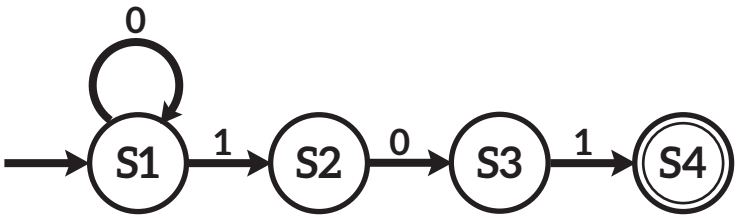
Computational Linguistics
- Sheet 1 -

Q1) What is the language recognized by the following FSA :

This accepts all strings
over 0 , 1 such that
any string ends with '101'

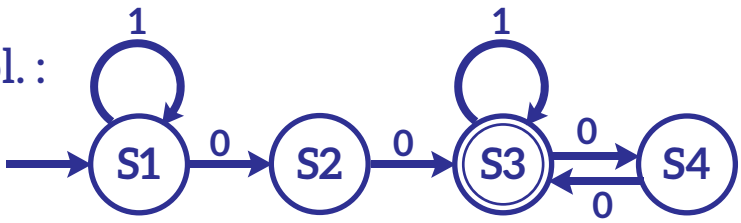


Q2) Draw a transition diagram for a FSA that accepts the language that consists of :

1- Strings containing the sequence 00 and Σ = { 0, 1}.
2- Strings containing even number of Zero's and Σ = { 0, 1}.
3- Strings with exactly 3 a's and Σ = { a, b}.
4- Strings where (number of a's mod 3) = (number of b's mod 3), and Σ = { a, b}. (e.x. abaabbaaa)
5- Strings that start and finish with 0 and Σ = { 0, 1}.
6- Binary strings of length 4 and Σ = { 0, 1}. Recognized strings would include 0000, 1111, 0101, but not 01, or 00000.

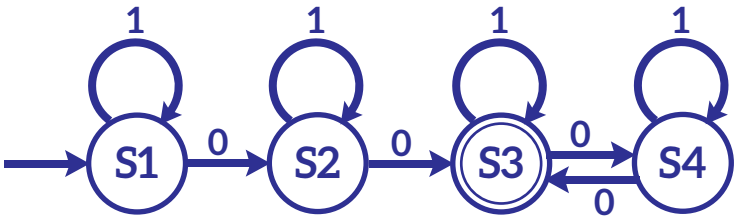
sol. :

1-



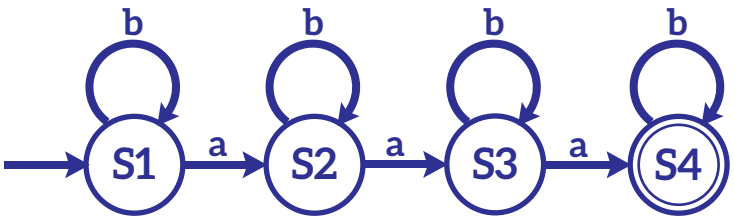
strings: 00 , 1001 , 100100 , 0000 , 00100 , ...

2-

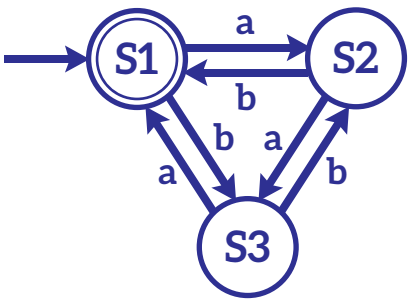


strings: 00 , 0000 , 100100 , 01010101 , ...

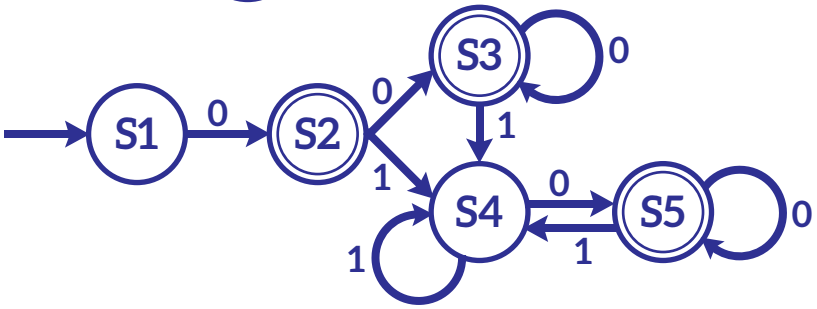
3-



strings: aaa , ababab , bbaaa , abbaa , ...

4- 

strings: abaabbaaa , aaa , bbb, abbbaa , ...

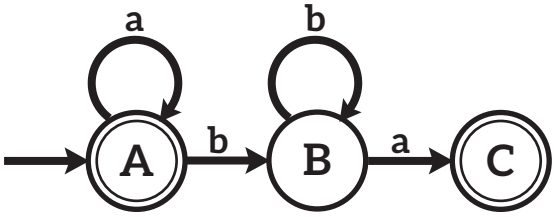
5- 

strings: 0 , 00 , 00100 , 0101010 , ...

6- 

strings: 1111 , 0000 , 0101 , ...

Q3) Given the following FSA with Σ = { a, b }



1- Write down two words that this machine accepts and two words that it does not accept.
2- Describe the language that this machine recognizes.
3- Write down the regular expression representing the language that this machine accepts.
4- Give the mathematical model for this machine -i.e. give:
- Q - the set of states,
- I - the start state,
- F - the set of final states,
- E - the set of rules that map a state to the next state.

sol. :
1-
- Accepts : ε , a , aa , ... , ba , bba , ... , aba , abba , ...
- Does not accept : b , bb , bbb , ... , bab , baab , ...

2- This accepts all strings over a , b such that any string ends with 'a'
3- Regular expression : **$a^* \mid a^*b^+a$**
4- • Q = { A , B , C }
   • I = { A }
   • F = { A , C }
   • E = { (A,a,A) , (A,b,B) , (B,b,B) , (B,a,C) }

Q4) What strings can be generated from the following regular expressions:

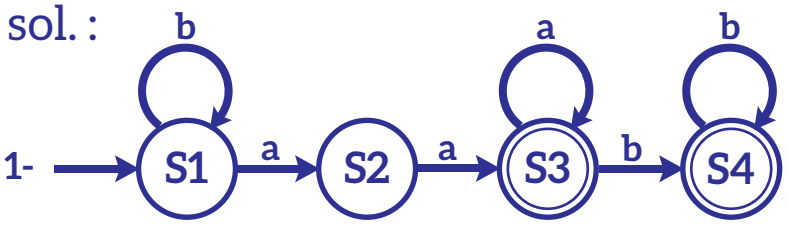
1- $(a^*b^*)^*$
2- $(a^+b)^*$
3- $(abc^*)^+$
4- $(a|b|c^*)^+$

sol. :
1- ε , a , aa , ... , b , bb , ... , ab , abb , aab , ... , ba , baa , bba , ...
2- ε , ab , aab , ... , abab , ababab , ... , aabaab , aaaabaaaaab , ...
3- ab , abc , abcc , ... , abab , abcab , ababc , abccccabcc , ...
4- a , b , ε , aa , ab , ac , ba , bb , bc , ca , cb , cc , ababccca , ...

Q5) Draw a graph for FSA that represents the following regular expressions:

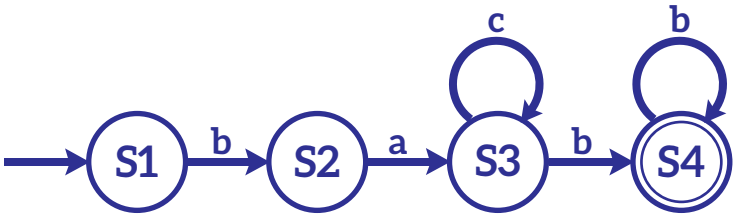
1- $b^*a^*(aa)a^*b^*$
2- bac^*b^+
3- $(a|c)^*|b$
4- a^+bc^+
5- $(a|c)^+|b^*$

sol. :



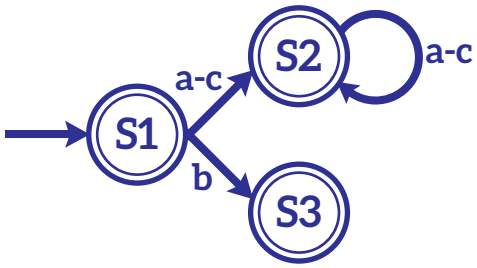
strings: aa , baab , baaaab , bbbbaab , ...

2-



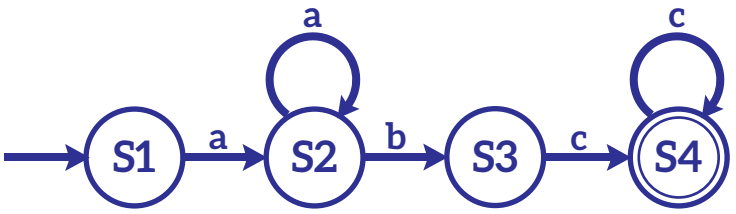
strings: bab , bacb , baccbb , babbbb , ...

3-



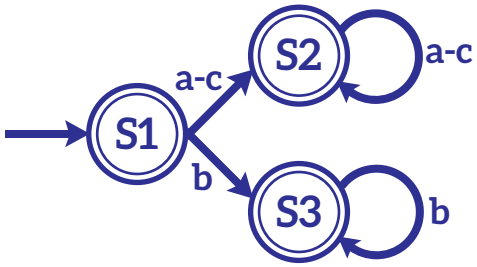
strings: ε , b , a , c , acacaca , ...

4-



strings: abc , aabcc , aaabccc , ...

5-



strings: ε , b , a , c , acaca , bbbb , ...

Q6) Given the following Context-Free Grammar, derive a valid sentence and draw its parse tree:

N : {S, NP, VP, DT, N, V}
S : {S}
Σ : {canary, cat, song, sings, eats, the}
P:
S → NP VP
NP → Dt N
VP → V NP
V → sings | eats
N → cat | song | canary
Dt → the

sol. :

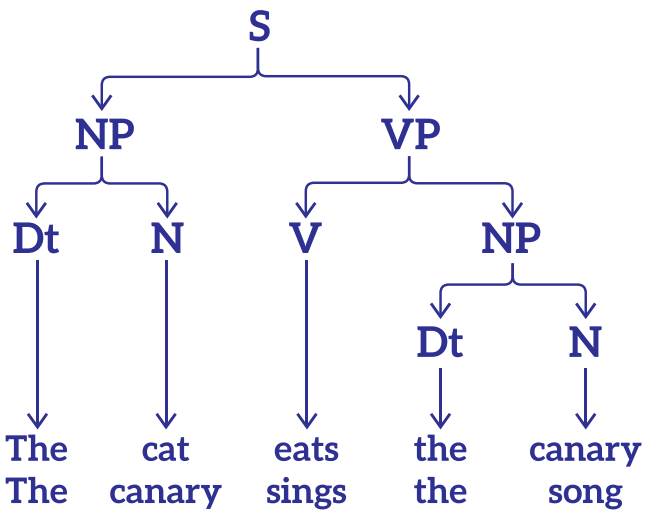
• Parsing:

S
NP VP
Dt N VP
The N VP
The cat VP
The cat V NP
The cat eats NP
The cat eats Dt N
The cat eats the N
The cat eats the canary

• Rule Used:

S → NP VP
NP → Dt N
Dt → the
N → cat
VP → V NP
V → eats
NP → Dt N
Dt → the
N → canary

• Parse Tree:



Q7) Given the following CFG:

N : {S, NP, NOM, VP, Det, Noun, Verb, Aux}
Σ : {that, this, a, the, man, book, flight, meal, include, read, does}
S : {S} , P:
S → NP VP
S → Aux NP VP
S → VP
NP → Det NOM
NOM → Noun
NOM → Noun NOM
VP → Verb
VP → Verb NP
Det → that | this | a | the
Noun → book | flight | meal | man | fish
Verb → book | include | reads
Aux → does

State if the following sentences are valid or not, and draw its parse tree:

1- "The man read this book"
2- "Book that Flight"
3- "Does this meal include a fish"

sol. :
1- Valid
2- Valid
3- Valid