POKHARA UNIVERSITY

Level: Bachelor | Semester: Fall | Year : 2023
Programme: BE | Full Marks: 100
Course: Theory of Computation | Pass Marks: 45
Time : 3 hrs.

Candidates are required to give their answers in their own words as far as practicable.

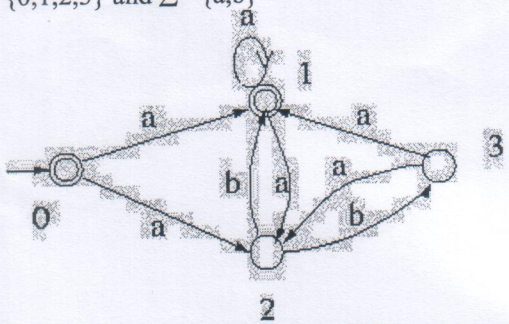
The figures in the margin indicate full marks.

Attempt all the questions.

1. a) Define Finite State Automata. Construct a DFA to recognize a language L that accepts the set of strings which contains neither aa nor bb as substring over $\sum$= {a, b} and also test your design with a valid string. 7

   b) What are Regular Expressions (RE)? Construct an NFA for the RE (a|b)*.abc.(a|b) 8

2. a) Convert the following NFA to its equivalent DFA where Q= {0,1,2,3} and $\sum$= {a,b} 7



   b) What is an empty language? Convert the following context-free grammar (CFG) into its equivalent Chomsky's normal form (CNF) 8

   i. $S \rightarrow a \mid aA \mid B$
   ii. $A \rightarrow aBB \mid \varepsilon$
   iii. $B \rightarrow Aa \mid b$

3. a) What is a Parse tree (Derivation tree)? How is it useful to show the grammar is ambiguous ? Give an Example. 7

   b) Explain PDA? Design a PDA which accepts the language L={ ww^R : wϵ $\{a,b\}^*$, w is a string and w^R represents reverse of w} and test for strings bbaabb and ababa. 8

4. a) Show that the language L={$a^n b^n c^n$: n>0} is not context free using the concept of pumping lemma. 7

b) "***For every CFG there is an equivalent Pushdown Automata***". Justify this statement with an example. 8

5. a) Design a Turing machine for computing function: 7
F (x, y) = x + y and show your validation for x = 2 and y = 4.

b) Briefly explain the idea of designing the Turing Machine that accepts the language $L = \{a^n b^n : n>0\}$. Show the state transition diagram for it. 8

6. a) What is computational complexity of a problem? Explain P, NP and NP-Complete problems. 7

b) Explain the Halting Paradox in Turing Machine. What are Space and Time complexity? 8

7. Write short notes on: (**Any two**) 2×5

a) Power set and Kleene Closure

b) Recursive and Recursively Enumerable Language

c) Universal Turing Machine