POKHARA UNIVERSITY

Level: Bachelor
Programme: BE
Course: Theory of Computation (New)

Semester: Spring

Year : 2024
Full Marks: 100
Pass Marks: 45
Time : 3hrs.

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

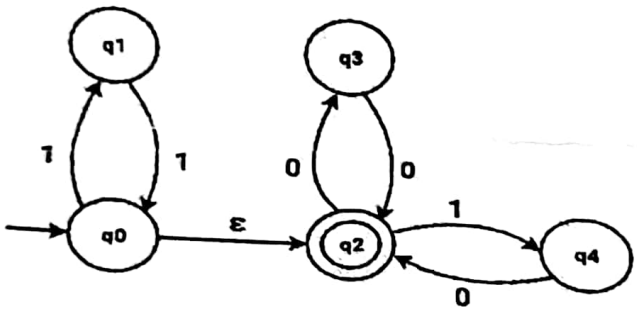
The figures in the margin indicate full marks.

Attempt all the questions.

1. a) State and prove the pigeonhole principle. Prove by mathematical induction that $n^2 - 3n + 4$ is even and true for all positive integers. 7

   b) Define Finite State Automata. Construct a DFA that recognizes Language L that accepts the set of strings that starts and ends with different symbols over $\Sigma=\{a, b\}$ and test your design with a valid string. 8

2. a) Convert the following NFA to its equivalent DFA. 7



   b) Describe the decision properties of regular languages. 8

3. a) What is CFL? Convert the given Grammar into CNF. 7

   S->ABAB A-> aA|ϵ B->bB|ϵ

   b) Design a PDA for the following language $L=\{a^n b^{2n+1} : n>0\}$ also check it for aabbbbbbb and aabbb. 8

OR

State the rules followed to design a PDA for a given CFG. Design a PDA that accepts $L= \{a^{3n} b^n : n>0\}$ and check the string aaaaaabb.

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) PDA is Stronger than FA and for every CFG there is an equivalent PDA. Justify this statement with an example. 8

5. a) Design a TM for L={WW^R: W^R is reverse of w ε (a,b)* for both even and odd palindrome}. 8

b) Differentiate between FA, PDA, and TM on the basis of string acceptance. Also, explain why does TM is considered functionally stronger among all of these? 7

OR

"Turing machine is believed to be the ultimate calculating mechanism", elaborate with the help of Church Turing thesis". Also, show that the function f(n)=n+2 is Turing computable.

6. a) Elaborate the properties of recursively enumerable and recursive languages. 8

b) Differentiate the time complexity of a Turing machine and time complexity of a language decision problem. Define the complexity classes *P* and *NP*. 7

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

a) Universal Turing Machine

b) Church-Turing Thesis

c) Halting Problem