2016/ODD/08/24/MCS–701 (C)/ MS–101 (C)/CS–701 (O)/593

PG Odd Semester (CBCS and Old) Exam., December—2016

COMPUTER SCIENCE

(7th Semester / 1st Semester)

Course No. : MCS–701 (C)/MS–101 (C)/CS–701 (O)

(Theory of Computation)

Full Marks : 75
Pass Marks : 30 (for CBCS)
Pass Marks : 26 (for Old Course)

Time : 3 hours

The figures in the margin indicate full marks for the questions

Answer **five** questions, taking **one** from each Unit

UNIT—I

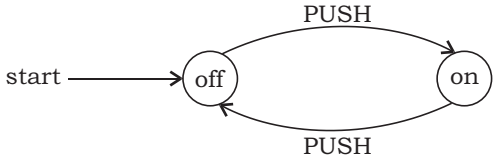
1. *(a)* State the contrapositive of the statement "If H then C".
State various cases to be considered to prove the logical equivalence of "If H then C" with its contrapositive. 3

(b) Given an alphabet $\{0, 1\}$. Construct ^{1}, ^{2}, ^{3} 2

(c) Construct the DFA which accepts the language such that
L $\{w|w$ has 011 as a substring$\}$
over the alphabet $\{0, 1\}$. Construct both the transition diagram and the transition table. 7

(d) Explain how the transition function of a DFA can be extended to accept strings. Use induction technique to prove that $\hat{}(q, w)$ $(\hat{}(q, x), a)$ for any string w xa. 3

2. *(a)* What is proof by mutual induction? Consider the following automata representing the operation of an on/off switch.

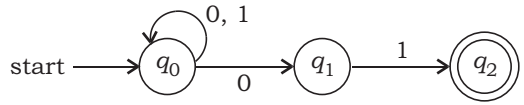


By using mutual induction technique prove the following two statements which explain the operation of the switch :

$S_1(n)$: The automata is in state "off" after n pushes if and only if n is even.

$S_2(n)$: The automata is in state "on" after n pushes if and only if n is odd. 6

(b) Consider the following NFA which accepts all strings ending with 01 for $\{0, 1\}$:



(i) Construct the transition table for the NFA.

(ii) Construct the equivalent DFA of the above NFA. 4

(c) Design an NFA to recognize the strings *abc*, *abd* and *aacd* for the alphabet $\{a, b, c, d\}$ and then convert your NFA to its equivalent DFA. 5

UNIT—II

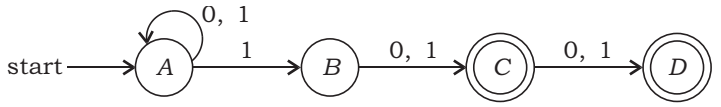
3. *(a)* Given the alphabet $\{a, b, c\}$, write regular expression for the set of strings containing at least one *a* and at least one *b*. 2

(b) The transition table for a DFA is given below : 7

	0	1
q_1	q_2	q_1
q_2	q_3	q_1
q_3	q_3	q_2

Convert the DFA to regular expressions.

(c) Consider the NFA (given below) which accepts strings having a '1' either two or three positions from the end :



(i) Convert the NFA to an automata with regular expression labels with no state elimination.

(ii) Perform all possible state eliminations on the above NFA to get the final regular expression for the entire automata. 6

4. *(a)* State and prove the pumping lemma for regular languages. 5

(b) Show that the language L_{pr} consisting of all strings of 1's where length is a prime is not a regular language. 4

(c) Prove that if L is a regular language, then the language L^R consisting of reversal of all the strings of L is also a regular language. 6

UNIT—III

5. *(a)* Define context free grammars (CFG) and write the production for the grammar G_{pal} for the palindromes given below over alphabet $\{0, 1\}$: 5

$$G_{pal} \quad (\{p\}, \{0, 1\}, A, P)$$

(b) Write the grammar for the language of regular expression

$$0\ 1(0\ \ 1)$$

and construct the parse tree for the following strings : 5

(i) 00101

(ii) 1001

(iii) 00011

(c) Convert the following grammar to Chomsky Normal Form (CNF) :

$E \quad E \quad T|T \quad F|(E)|a|b|Ia|Ib|I0|I1$

$T \quad T \quad F|(E)|a|b|Ia|Ib|I0|I1$

$F \quad (E)|a|b|Ia|Ib|I0|I1$

$I \quad a|b|Ia|Ib|I0|I1$ 5

6. *(a)* What is push-down automata? Design a push-down automata along with its transition diagram to accept the language L_{wwr} given by

$$L_{wwr} \quad \{ww^R \mid w \text{ is in } (0\ \ 1)\ \}$$

which is a language called "ww reversal" and is the even length palindromes over the alphabet $\{0, 1\}$. 3+5=8

(b) Explain the pumping lemma for context free languages. 7

UNIT—IV

7. *(a)* What is a turing machine? Explain the conceptual foundation of turing machine. 3

(b) State the formal definition of turing machine and explain its components. 5

(c) Design a turing machine to accept the language $\{0^n\ 1^n \mid n \quad 1\}$ and explain its behaviour. 7

8. *(a)* What is instantaneous description (ID) for turing machine? Explain. 6

(b) Design a turing machine to accept palindromes. 5

(c) Explain the concept of 'Recursively Enumerable Languages'. 4

UNIT—V

9. *(a)* What do you mean by 'undecidable' problems? Discuss the reduction of an undecidable problem to another problem. 7

(b) What do you mean by non-deterministic turing machine? Prove that if M_N is a non-deterministic turing machine, then there is a deterministic turing machine M_D such that

$$L(M_N) \quad L(M_D)$$ 8

10. *(a)* Define universal languages. Explain the organization of a universal turing machine. 10

(b) Discuss the issues of computational complexity for a turing machine simulated on a computer. 5

★ ★ ★