G 1580 (Pages : 3) Reg. No.....

Name.....

B.TECH. DEGREE EXAMINATION, MAY 2016

Fourth Semester

Branch : Computer Science and Engineering/Information Technology

CS 010 406/IT 010 404—THEORY OF COMPUTATION (CS/IT)

(New Scheme—2010 Admission onwards)

[Regular/Improvement/Supplementary]

Time : Three Hours Maximum : 100 Marks

Part A

Answer **all** *questions.*
Each question carries 3 *marks.*

1. What is diagonalization principle.
2. Write the formal definition of regular expressions.
3. What are null productions ? How they can be removed ?
4. Write the formal definition of Turing Machine.
5. Differentiate tractable and intractable problems.

(5 × 3 = 15 marks)

Part B

Answer **all** *questions.*
Each question carries 5 *marks.*

6. Prove that $f(x) = 2 * x$ and $f(x) = 2x$ are primitive recursive functions.
7. Write regular expressions for :
   (a) Set of all strings that end in double letter over {a, b}.
   (b) Set of all three lettered words starting with 'b' over {a, b}.
8. Define CFG. Give CFG for :
   (a) Strings with equal no of a's and b's.
   (b) Regular expression (011 + 1)* (01)*.

Turn over

9. What are multi-head and multi-tape Turing machines.

10. Explain the technique of polynomial time reduction.

(5 × 5 = 25 marks)

Part C

Answer **all** *questions.*

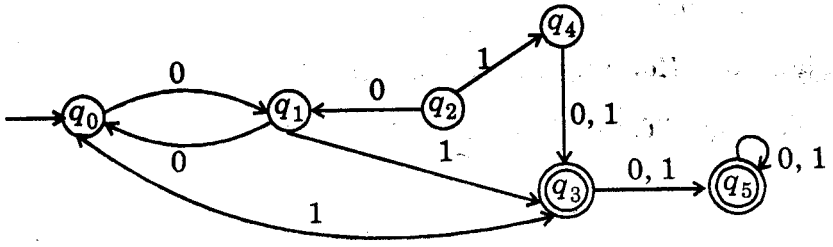
Each full question carries 12 *marks.*

11. Explain the proof by induction and prove that for a finite set A, $| 2^A | = 2^{|A|}$.

Or

12. Explain Chomsky classification of languages.

13. Minimize the given DFA. Explain the algorithm



Or

14. Explain different applications of finite automata.

15. Design a PDA for the language $L = \{ w \mid w \in (a,b)^* ;\ n_a(w) > n_b(w) \}$.

Or

16. Explain simplification of CFG by simplifying the given CFG :

$G = (\{s, x, z, c\}, \{0, 1\}, p, s)$

$P : s \to 0x \mid 0\,11, x \to 00x \mid \in$

$z \to \mid z \mid 11c, c \to z.$

17. (a) Construct a turing machine for $L = \{a^n, b^n, c^n, n \geq 0\}$ and show the trace for one acceptable and one non-acceptable strings.

(8 marks)

(b) Design a turing machine for $f(n) = n \bmod 2$.

(4 marks)

Or

18. Explain universal turing machine in detail.

19. Explain various complexity classes with proper examples.

Or

20. Prove that 'clique' problem is NP-complete.

(5 × 12 = 60 marks)