G 1708 (Pages : 2) Reg. No.....

Name.....

B.TECH. DEGREE EXAMINATION, JULY/AUGUST 2007

Seventh Semester

Branch—Computer Science and Engineering

THEORY OF COMPUTATION (R)

(2002 Admissions)

Time : Three Hours Maximum : 100 Marks

Answer **all** *questions.*

Part A

1. Define primitive recursive function.
2. Explain composition of functions.
3. Define regular expression.
4. How will you use pumping lemma to show that certain languages are not regular ? Give the general steps.
5. When do you call a context free grammar ambiguous ? Give an example.
6. Give a CFG that generates all palindromes over $\{a, b\}$ and show that derivation of *ababa*.
7. Define Turing machine.
8. Briefly explain Church's thesis.
9. Give the general steps of proving by means of reduction that certain problems are undecidable.
10. When do you call a decision problem NP-hard ?

(10 × 4 = 40 marks)

Part B

11. (a) Prove that the function $f_{add}(x, y) = x + y$ is primitive recursive. (8 marks)

    (b) Prove that the function $g(x, y) = x^y$ is primitive recursive. (4 marks)

Or

12. (a) Show that the set of real numbers is not denumerable. (6 marks)

    (b) Explain the significance of Theory of computation in computer science. (6 marks)

13. (a) Prove that $L = \{ww^R \mid w \in \{a, b\}^*\}$ is not regular. (6 marks)

    (b) Prove that $L = \{a^p \mid P \text{ is a prime number}\}$ is not regular. (6 marks)

Or

14. (a) Give the regular expression for the language $L = \{a^n b^m \mid n \geq 4, m \leq 3\}$ and its component L'. (6 marks)

    (b) Define DFA and construct a DFA that accepts $L = \{w \in \{a, b\}^* \mid$ no. of a's in w is even and no. of b's in w is odd$\}$. (6 marks)

Turn over

→ useful in analysis of which problems can be efficiently solving on parallel computers.

→ A decision problem is in P-Complete if it is Complete for P, meaning that it is in P, and that every problem in P can be reduced to it in polylogarithmic time on parallel computer with a polynomial no. of processors.

CVP – Circuit Value pblm – given a ckt, the i/ps to the ckt, and one gate in the ckt, calculate the o/p of that gate.

15. (a) Construct a PDA which accepts that language $\{wcw^R \mid w \in \{a. b\}^*\}$ and show the processing of *abcba*. (7 marks)

(b) Briefly explain parsing using PDA with an example. (5 marks)

Or

16. (a) Construct a PDA which accepts the language $\{ww^R \mid w \in \{a, b\}^*\}$ and show the processing of *abba*. (7 marks)

(b) Design a CFG for the language $L = \{a^n b^m \mid n$ not equal to $m\}$. (5 marks)

17. (a) Construct a Turing machine that increments a binary number. (8 marks)

(b) Write short notes on any *two* variants of Turning machine. (4 marks)

Or

18. (a) Construct a Turing machine that decrements a binary number. (8 marks)

(b) Construct a Turing machine that adds two unary numbers. (4 marks)

19. State Turing machine halting problem. Show that it is undecidable.

Or

20. Show that the problem of checking whether a graph has a clique of size k is NP-complete. (12 marks)

[5 × 12 = 60 marks]

c) Decision problem is P – Complete if it is in P and that every problem in P can be reduced to it by using an appropriate reduction.

eg. – The most basic P-Complete problem is this: Given a Turing machine, an i/p for the machine, and a number T (written in unary), does that machine halt on that input within the first T steps. It is clear that it is P-Complete. If we can parallelize a general simulation of a sequential computer, then we will be able to parallelize any pgm that runs on the computer.

eg. – TSP, Cycle Covers, 0-1 integer pgmg, ... quadratic assignment