

G 1086

(Pages : 2)

Reg. No.....

Name.....

B.TECH. DEGREE EXAMINATION, MAY 2016

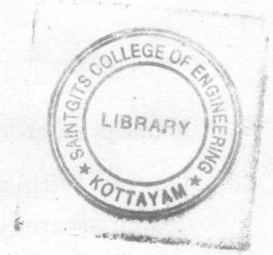
Sixth Semester

Branch : Computer Science and Engineering

ALGORITHM ANALYSIS AND DESIGN (R)

(Old Scheme—Prior to 2010 Admissions)

[Supplementary/Mercy Chance]



Time : Three Hours

Maximum : 100 Marks

Part A

Answer all questions.

Each question carries 4 marks.

1. Define an algorithm. List the properties of an algorithm.
2. What is pseudo code? Write pseudo code to accept two numbers, compute their sum and print the result.
3. How divide-and-conquer design strategy works? Explain.
4. Explain how binary search works.
5. Why only a few optimization problems can be solved by the greedy method? Discuss.
6. What is a minimum cost spanning tree? Give example.
7. State and explain the principle of optimality.
8. Explain the travelling salesman problem with an example.
9. What is backtracking? Explain.
10. State and explain the N-Queens problem.

(10 × 4 = 40 marks)

Part B

Answer all questions.

Each question carries 12 marks.

11. (a) Compute the big-Oh running time of the following C code segment :

```
for i = 1 ; i < n ; i++
{
sum = sum + i ;
}
```

(4 marks)

Turn over

- (b) Explain with example how recurrence relations can easily describe the runtime of recursive algorithms.

(8 m)

Or

12. What is a deterministic algorithm? How deterministic algorithms work? Discuss with an example.
13. Discuss with an example Strassen's matrix multiplication algorithm using divide-and-conquer design strategy.

Or

14. Explain the merge sort algorithm to sort an array of 'N' numbers in ascending order. Illustrate each step of the algorithm with an example.
15. How a fractional knapsack problem can be solved using a greedy strategy? Explain with algorithm and example.

Or

16. Explain with an example Kruskal's algorithm for constructing a minimum cost spanning tree.
17. What type of problems dynamic programming is used to solve? How dynamic programming is used to solve the all-pairs shortest path problem? Discuss with an algorithm and example.

Or

18. What are comparison trees? Explain with an algorithm and example the use of comparison trees for sorting.
19. How backtracking can be used for the knapsack problem? Explain with an algorithm and example.

Or

20. Discuss the least cost branch-and-bound search algorithm with example and diagrammatic illustrations.

(5 × 12 = 60 m)

