

[58/A-27]

SEAT No. \_\_\_\_\_

No. of printed pages: 2

Sc

**SARDAR PATEL UNIVERSITY**  
**Master of Computer Applications (MCA)**  
Semester – III External Examinations  
**PS03CMCA23 – Analysis and Design of Algorithms**  
Saturday, 24<sup>th</sup> November, 2018

Time: 02:00 p.m. to 05:00 p.m.

Max Marks: 70

**Q1. Choose the most appropriate option for each question.**

**[8]**

- i. \_\_\_ Pseudo code notation is used to indicate assignment statement, while writing algorithm.  
(A) = (C) ==  
(B) := (D) All of these
- ii. While measuring performance of an algorithm, the space and time needed for compilation is \_\_\_  
(A) Not considered (C) Considered  
(B) Optional (D) Averaged
- iii. Knapsack problem fits in the category of \_\_\_  
(A) Ordering Paradigm (C) Both (A) & (B)  
(B) Subset Paradigm (D) None of these
- iv. \_\_\_ is a Boolean valued function that determines whether x can be included into the solution or not, in greedy method.  
(A) Select (C) Project  
(B) Union (D) Feasible
- v. Tree organizations which are problem instance dependent are called \_\_\_\_\_.  
(A) Hybrid trees (C) Dynamic trees  
(B) Static trees (D) None of these
- vi. \_\_\_\_\_ is a generated node which is to be expanded further.  
(A) Dead node (C) Live node  
(B) E-node (D) None of these
- vii. \_\_\_\_\_ are those problem states s for which the path from the root to s defines tuple in the solution space.  
(A) Answer states (C) Both (A) & (B)  
(B) Solution states (D) None of these
- viii. \_\_\_\_\_ are rules that restrict each xi to take on values only from a given set.  
(A) Implicit Constraints (C) Both (A) & (B)  
(B) Explicit constraints (D) None of these

**Q2. Answer the following questions (Any Seven):**

**[14]**

- a. List types of recursive algorithms.
- b. What is space complexity of an algorithm? Which are two components of it?
- c. Briefly explain the concept of tree.
- d. Compare Heapify and Insert algorithms briefly.
- e. What do you mean by "Feasible Solution" of a problem?
- f. Define: Principle of Optimality.
- g. Give an example of sum of subset problem.

(P.T.O.)

- h. What is Hamiltonian cycle? Give an example.
- i. What is 0/1 Knapsack problem?

**Q3. Answer the following questions:**

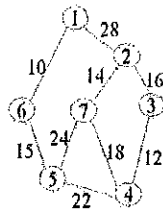
- a. What is Algorithm? Explain criteria to design an algorithm. [6]
- b. What is asymptotic notation? Define all asymptotic notations; explain any one of them in detail. [6]

**OR**

- b. Write an algorithm for finding maximum and minimum element in given set using divide and conquer method. [6]

**Q4. Answer the following questions:**

- a. Explain the knapsack problem in detail. Write the algorithm to solve this problem. [6]
- b. Find the minimum cost spanning tree for following graph G, using Prim's method. Show all intermediate steps. [6]



**OR**

- b. Explain the logic of Dijkstra's algorithm. Show the execution of Dijkstra's algorithm by giving example of your choice. [6]

**Q5. Answer the following questions:**

- a. Write an algorithm of multistage graph problem (Forward approach). Explain it taking suitable example. [6]
- b. Explain n-queens problem taking suitable examples. [6]

**OR**

- b. Write algorithm of Hamiltonian cycle problem. Explain it taking suitable example. [6]

**Q6. Answer the following questions:**

- a. Write and explain an algorithm of Least Cost search. [6]
- b. Explain the 15-puzzle problem taking suitable example. [6]

**OR**

- b. Consider the travelling salesperson instance defined by the following cost matrix: [6]

$$\begin{pmatrix} \infty & 8 & 3 \\ 7 & \infty & 5 \\ 4 & 6 & \infty \end{pmatrix}$$

- (i) Obtain the reduced cost matrix.
- (ii) Generate its state space tree using Least Cost Branch-and-Bound.