

Oct-22/BE/Insem-41

B.E. (Computer Engineering)**DESIGN AND ANALYSIS OF ALGORITHMS**(2019 Pattern) (Semester - VII) (410241)

Time : 1 Hour]

[Max. Marks : 30]

Instructions to the candidates:

- 1) Answer the question of Q.1 or Q.2, Q.3 or Q.4.
- 2) Neat diagrams must be drawn whenever necessary.
- 3) Figures to the right indicate full marks.
- 4) Assume suitable data, if necessary.

- Q1)** a) Why correctness of the algorithm is important? Define loop invariant property and prove the correctness of finding summation of n numbers using loop invariant property. [8]
 b) What is iterative algorithm? Explain iterative algorithm design issues using examples. [7]

OR

- Q2)** a) How to prove that an algorithm is correct? How to prove the correctness of an algorithm using counter example? Give suitable example. [7]
 b) Write a short note on any 4 problem solving strategies. [8]

- Q3)** a) What is Best, Average and Worst case Analysis of Algorithms? Analyse the following algorithm Best, Average and Worst case
 void sort (int a. int n){

```

    int i, j;
    for (i = 0; i < n; i++) {
        j = i-1;
        key = a[i];
        while (j >=0 && a[j] > key)
        {
            a[j+1] = a[j];
            j = j-1;
        }
        a[j+1] = key;
    }
}
```

P.T.O.

- b) • Explain P, NP, NP-Hard and NP-Complete problems with examples.
- Explain 3-SAT problem using an example. Why is SAT so important in theoretical computer science?

[7]

OR

- Q4) a) What is NP-complete class problem? How would you prove vertex cover problem is NP-complete class problem? [8]
- b) What is Best, Average and Worst case Analysis of Algorithms? Analyse the following algorithm Best, Average and Worst case [7]

```
int Linear-search(int a, int n, int item) {  
    int i;  
    for (i = 0; i < n; i++) {  
        if (a[i] == item) {  
            return a[i]  
        }  
    }  
    return -1  
}
```