A.Y: 2021-22

Department of Computer Science and Engineering

BAPATLA ENGINEERING COLLEGE, BAPATLA

(AUTONOMOUS)

First Assignment Examination

Class: **II/IV B.Tech.(CSE/CS/DS) Sec A,B,C1stSem** Max.marks:**10m.**

Subject: **DAA** Time: **45mins.**

Date: **05/05/2022**

PART-A (6M)

1. Define the term algorithm and explain about criteria of an algorithm
2. Express the return value of the function "mystery" in.

```
Int mystery(int n)
 {
        int j=0, total=0;
        for (int i=1;j<=n;i++)
        {
                ++total;
                j+=2*i;
        }
        return total;
    }
```

B. Consider the following C function

```
Int check(int n)
{
        inti,j;
        for (i=1;i<=n;i++)
        {
                for (j=1;j<n;j+=i)
                {
                printf("%d",i+j);
                }
        }
}
```

Find the time complexity of check in terms of theta notation

3. A. Solve $T(n)=7T(n/2)+n^2$ if n>2

   =1 if n=2 Using iteration method.

   B. Arrange the increasing order of asymptotic complexity of functions f1, f2, f3 and f4?

$$f1(n) = \mathbf{2^{\wedge n}}\ ,\ f2(n) = \mathbf{n^{\wedge(3/2)}}\ ,\ f3(n) = \mathbf{nlogn}\ ,\ f4(n) = \mathbf{n^{\wedge(logn)}}$$

4. A. Explain about specifications of an algorithms.
   B. list out all three cases for master method by using divide and conquer,
   . Solve (master's method)
   i. $T(n)=7T(n/2)+n^2$
   ii. $T(n)=2T(n/2)+nlogn$
5. Define asymptotic notation. Explain all asymptotic notations used in algorithm analysis.
6. Find out space complexity

   A.
```
Algorithm(A,1,n)
{
        int i,B[ ];
        for i := 1 to n
        {
                A[i]=10;
                B[i]=A[i];
        }
 }
```

Department of Computer Science and Engineering

BAPATLA ENGINEERING COLLEGE, BAPATLA

(AUTONOMOUS)

First Assignment Examination

Class: **II/IV B.Tech.(CSE/CS/DS) Sec A,B,C1stSem** Max.marks:**10m.**

Subject: **DAA** Time: **45mins.**

Date: **05/05/2022**

B. Algorithm A(n)

```
{
        if(n>=1)
        {
                A(n-1);
                Printf("n");
                A(n-1);
        }
}
```

PART-B(1M)

1. Define time complexity.
2. Define the terms Best case, Worst case and Average case time complexities.
3. Define Bog-O notation.
4. What is amortized analysis.
5. If $f(n)=5n^2 + 6n + 4$, then prove that T(n) is $O(n^2)$
6. What is the time complexity for Fibonacci series?
7. What is the recurrence relation for Towers of Hanoi problem?
8. What is the efficiency of algorithm?
9. Give the two major phases of performance evaluation.
10. What are the basic asymptotic efficiency classes?
11. List the factors which affects the running time of the algorithm.
12. Define Space complexity.
13. Compare **n** , $\mathbf{(logn)^{100}}$ which one is large.
14. Find out time complexity If $f(n)=n!+nlogn+n^{50}$
15. What is the order of growth?