Design and Analysis of Algorithms

Tutorial-2

1. What is the time complexity of below code and how?

```
void fun(int n){
  int j = 1, i = 0;
  while (i < n){
     i = i + j;
     j++;}}
```

2. Write recurrence relation for the recursive function that prints Fibonacci series. Solve the recurrence relation to get time complexity of the program. What will be the space complexity of this program and why?

3. Write programs which have complexity – n(logn), n^3, log(logn)

4. Solve the following recurrence relation T(n) = T(n/4) + T(n/2) + cn^2

5. What is the time complexity of following function fun()?

```
int fun(int n){
   for (int i = 1; i <= n; i++){
      for (int j = 1; j < n; j += i){
         // Some O(1) task
      }}}
```

6. What should be the time complexity of

```
         for (int i = 2; i <=n; i = pow(i, k))
         {
            // some O(1) expressions or statements
         }
```

where, k is a constant.

7. Write a recurrence relation when quick sort repeatedly divides the array in to two parts of 99% and 1%. Derive the time complexity in this case. Show the recursion tree while deriving time complexity and find the difference in heights of both the extreme parts. What do you understand by this analysis?

8. Arrange the following in increasing order of rate of growth:

a) n, n!, logn, loglogn, root(n), log(n!), nlogn, log^2(n), 2^n, 2^(2^n), 4^n, n^2, 100

b) 2(2^n) , 4n , 2n , 1 , log(n) , log(log(n)) , √log(n) , log2n , 2log(n) , n , log(n!) , n! , n2 , nlog(n)

c) 8^(2n) , log$_2$(n) , nlog$_6$(n) , nlog$_2$(n) , log(n!) , n! , log$_8$(n) , 96 , 8n2 , 7n3 , 5n