Design and Analysis of Algorithms: Assignment-1

Designed By – Akshay Rajput (Assistant Professor CSE GEU)

(Max Marks – 05)

Please follow the below mentioned instructions:

1. Attempt assignment in a separate thin notebook or A4 sheets.

2. Deadline to submit the assignment is 17th March. No submission will be entertained after that.

3. Avoid duplicating the code. If you need to write same pseudocode or program at multiple places, I recommend merging all such problems.

4. Write point to point answers. Avoid writing unnecessary stories.

5. Use standard reference book, geeksforgeeks or stackoverflow for answering all the problems.

1. What do you understand by Asymptotic notations. Define different Asymptotic notation with
examples.

2. What should be time complexity of – for(i=1 to n) {i=i*2}

3. T(n) = {3T(n-1) if n>0, otherwise 1}

4. T(n) = {2T(n-1)-1 if n>0, otherwise 1}

5. What should be time complexity of -

```
int i=1, s=1;
while(s<=n){
        i++; s=s+i;
        printf("#");
}
```

6. Time complexity of -

```
void function(int n){
        int i, count=0;
                for(i=1; i*i<=n; i++)
                        count++
}
```

7. Time complexity of -

```
void function(int n){
        int i,j,k, count=0;
        for(i=n/2; i<=n; i++)
                for(j=1; j<=n; j=j*2)
                        for(k=1; k<=n; k=k*2)
                                count++
}
```

8. Time complexity of -

```
function(int n){
        if(n==1) return;
        for(i=1 to n){
                for(j=1 to n){
                        printf("*");
                }
        }
        function(n-3);
}
```

9. Time complexity of –

```
void function(int n){
        for(i=1 to n){
                for(j=1; j<=n; j=j+i)
                        printf("*")
        }
}
```

10. For the functions, n^k and a^n , what is the asymptotic relationship between these functions?
Assume that K>=1 and a > 1 are constants. Find out the value of c and n_0 for which relation holds.

11. What is the time complexity of below code and why?

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

12. 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?

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

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

15. 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
                }
        }
}
```

16. 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.

17. 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?

18. Arrange the following in increasing order of rate of growth:
a) n, n!, logn, loglogn, root(n), log(n!), nlogn, 2^n, 2^{2n}, 4^n, n^2, 100
b) $2(2^n)$, 4n , 2n , 1 , log(n) , log(log(n)) , $\sqrt{}$log(n) , log2n , 2log(n) , n , log(n!) , n! , n^2 , nlog(n)
c) 8^(2n) , $\log_2(n)$, $n\log_6(n)$, $n\log_2(n)$, log(n!) , n! , $\log_8(n)$, 96 , $8n^2$, $7n^3$, 5n

19. Write linear search pseudocode to search an element in a sorted array with minimum comparisons.

20. Write pseudo code for iterative and recursive insertion sort. Insertion sort is called online sorting. Why? What about other sorting algorithms that has been discussed in lectures?

21. Complexity of all the sorting algorithms that has been discussed in lectures.

22. Divide all the sorting algorithms into in place/stable/online sorting.

23. Write recursive/iterative pseudo code for binary search. What is the Time and Space complexity of Linear and Binary Search (Recursive and Iterative)

24. Write recurrence relation for binary recursive search.