Nirma University
Institute of Technology
B. Tech CSE Sem. V
2CS503 – Design and Analysis of Algorithms
Class Test 1, Aug 2021

Total Marks: 35 Time: 75 minutes

Roll No. | Supervisor's initial with date

Instructions:
1. Attempt all questions.
2. Figures to right indicate full marks.
3. Draw neat sketches wherever necessary.
4. Assume suitable data wherever necessary and specify clearly.

Q 1 CO1 Solve/Obtain the total solution for the following recurrences. [12]

(a) $T(n) + 5T(n-1) + 6T(n-2) = 3n^2 - 2n + 1$

(b) $t_n = \begin{cases} 0 & if\ n = 0 \\ 2t_{n-1} + n + 2^n & otherwise \end{cases}$

Q 2 CO1 Show that if $f(n) = 100*2^n + n^5 + n$ then $f(n) = O(2^n)$. Report c and n_0. [4]

Q 3 CO1 Consider the following function : [8]

```
int SequentialSearch(int A[], int &x, int n)
{
int i;
  for(int i=0; i<n && a[i]!=x; i++)
  {
       if(i==n)
            return -1;
  }
return i;
}
```

Derive the expression for T(n) providing detailed analysis considering cost of each instruction. Report and discuss best-case, average-case and worst-case time complexity.

Q 4 CO1 Consider the following code fragment where n denotes the input size. [2]

```
int fun(int n)
{
int i;
  for(i=1; i<=n; i++)
       printf("hello world");
}
```

Analyse the time complexity of the above code fragment in terms of O-notation. Justify your answer

Q 5 Solve the following recurrence using Master Method. If you can't solve a specific recurrence using a master method, justify the reason behind the same. [9]

(a) $T(n) = 4T(n/2) + n^2 \times \sqrt{n}$, where n denotes the input size.

(b) $T(n) = 3T(n/3) + n\log n$

(c) $T(n) = 2T(n/2) + n^2$