Semester: V (Regular)
Sub & Code: DAA, CS-3001
Branch (s): CSE & IT

AUTUMN MID SEMESTER EXAMINATION-2015

Design & Analysis of algorithm

[CS-3001]

Full Marks: 25 **Time: 2 Hours**

Answer any four questions including question No.1 which is compulsory.
The figures in the margin indicate full marks.
Candidates are required to give their answers in their own words as far as practicable and all parts of a question should be answered at one place only.

Q1 Answer the following questions: (2 x 5)

a) Consider the following C function.

```
int fun(int n)
{
int i, j, p=0;
for(i = 1; i < n; ++i)
for(j = n; j > 1; j = j/2)
++p;
return p;
}
```

What is most closely approximates value returned by the function fun?

b) Rank the following functions by order of growth in increasing sequence?
$\log n$, $\log \sqrt{n}$, $\sqrt{n}$, $n \log \sqrt{n}$, $n!$, 2^n

c) Can the master method be applied to solve the following recurrence?
$T(n) = 4T(n/2) + n^2 \log n$
Justify your answer.

d) Write the merge sort procedure (only) which divides the array into two parts such that first part contains elements twice of second part. Also derive the time complexity of that merge sort.

e) Match the following Algorithms and its recurrences

Bubble-sort	$T(n)=T(n/2)+ \Theta(1)$
Quick-sort	$T(n)=T(n-1)+ \Theta(n)$
Merge-sort	$T(n)=T(k) + T(n-k)+\Theta(n)$
Binary-Search	$T(n)=2T(n/2)+ \Theta(n)$

Q2 a) What is the significance of asymptotic notations? Define different asymptotic notations used in algorithm analysis. (2.5)

b) Solve the following recurrence. (2.5)
$T(n) = 4T(n/2) + n^2$ where $n>1$ and $T(1)=1$

Q3 a) Write the PARTITION() algorithm of Quick Sort and describe step by step how you would get the pass1 result by taking last element as pivot on the following data. (3)

8, 2, 1, 5, 6, 1, 3, 7, 4, 9, 5

Derive the average case time complexity of quick sort.

b) Given 10 activities, A=< a1, a2,….,an > along with their start time (si) and finish time (fi) as Si=< 1, 2, 4, 3, 7, 7, 8, 9, 11, 12 > and fi=< 3, 5, 4, 7, 5, 9, 14, 18, 10, 12 > and that requires the exclusive use of a common stage for scheduling these activities. Use an efficient method that computes a schedule with largest number of activities on that stage. (2)

Q4 a) Find an optimal solution to the knapsack instance n=7, W=15. (v1, v2, v3, v4, v5, v6, v7) = (10, 5, 15, 7, 6, 18, 3) and (w1, w2, w3, w4, w5, w6, w7) = (2, 3, 5, 7, 1, 4, 1), where n is the number of items, W is the knapsack capacity that thief can carry, v_i stands for value or profit w_i stands for weight of the i^{th} element. (2)

b) Sort the following data in descending order by using heap sort technique. (2)
50, 12, 28, 100, 18, 35, 75, 86, 15, 80

c) Consider a complete binary tree where the left and the right subtrees of the root are max-heaps. What is upper bound to convert the tree onto a heap? (1)

Q5 a) Write an algorithm to find out two elements from an array having non-negative numbers such that the difference should be maximum. (3)

c) An unordered list contains n distinct elements. What is the minimum number of comparisons required to find an element in this list that is neither maximum nor minimum? (2)

A. O(1) B. O(log n) C. O(n log n) D. O(n)

==