Practice Problems (Common to G1 and G2 Slots)

1. Let A[1…n] be an array of n integers. Using the divide and conquer approach write an algorithm that will find the element(s) that is not repeated. For eg. If the array A consists of elements 2,3,4,3,4,2,3,5,2,3 then the output will be '5'.
2. Look at the following pseudocode and answer the questions given below:

```
recurrentMethod( int[] a, int low, int high, int goal )
{
        int target = arrangeTarget( a, low, high );
        if ( goal < a[target] )
                return recurrentMethod( a, low, target-1, goal );
        else if ( goal > a[target] )
                return recurrentMethod( a, target+1, high, goal );

            else return a[ target ];

}
```

   a. What can you say about the best, average and worst case time complexity of the algorithm? Analyze each case separately with appropriate examples.
3. Let A[1…n] be an array of n integers and $k \in \{2,3,4,...9\}$ be an integer. Using the divide and conquer approach, find the sum of all numbers in A that are divisible by k. Derive the time complexity of your algorithm.
4. Let A[1…n] be an array of integers. Write an algorithm that will output all those pairs of numbers (A[i],A[j]) such that |A[i] – A[j]| < k for some $k, 1 \le k \le 9$.

5. Write an algorithm and subsequent C code for each following operations on binary tree. Assume nodes of the binary tree has positive integers only. -1 in the array indicates that node is not presented in the tree
   a. Number of leaf nodes
   b. Degree of each node. The degree of a node u is the number of nodes which are adjacent to u. For eg. In the figure above, degree of node A is 2, Degree of node B is 2 and degree of node C is 3.
6. Write a recursive algorithm using the divide and conquer approach and a subsequent C code for your algorithm to print the maximum digit of the given number x . For example, 8 is the maximum digit of 286883.
7. Let 'A' be an array of size 'n' and it contains ' n ' numbers. Write an algorithm (using divide and conquer technique) to sort only odd indices of the array A. Assume that the indices of the array 'A' starts from 1. For example, if A=

-1	2	-2	-5	4	0

is an input to your algorithm then your algorithm should give the following output

-2	2	-1	-5	4	0

8. A sample of radioactive substance is expected to decay by 0.15 percent each hour. What is the weight w(t) of the sample at t hours later in the experiment, where w(0) = 100 is the initial weight. Design a recursive algorithm to find the weight after 7 hours.
9. Write an algorithm using the divide and conquer approach and a brute force approach to find the number of different elements in an integer array. What will be the time complexity of both the algorithms? Which approach is better?
10. Let A be an nxn @D matrix such that the elements of the principle diagonal are all 0's and the remaning elements are all positive integers. Design an efficient divide and conquer strategy and subsequently write an algorithm to arrange the matrix A such that all the even numbers are replaced with the -1.
11. A Binary Search Tree has the property that, at each given node(current node), the values of the entries in the left subtree(if it exists) of the current node, is always less than or equal to the value of the entry at the current node, and the values of the entries in the right subtree(if it exists) of the current node is always greater than or equal to the value of the entry in the current node. Write an algorithm that will test if a given binary tree is a binary search tree or not.
12. **This question is tough and might probably not be suitable to ask in an examination, however since this is a practice, it would be a good exercise for you to solve.** Can you write an algorithm that will convert a Binary Max-Heap into a Binary Search Tree and vice-versa? A Max-Heap has the property that the key value of the parent node is greater than or equal to the key value of the child nodes. Analogously a Min-Heap has the property that the key value of the parent node is less than the key values of the child nodes.
13. Let T be a binary tree where each node i in T has a key value x_i, that is, if T is represented as an Array A, the the node i (indexed in the array as I in A) has key value A[i]. Let r be the root node of T. Clearly $r = i = 0$ Write an algorithm to find the sum of the key values of all nodes on the path from the root node r $(i = 0)$ to a given leaf node j. Denote this sum as $sum(r, j)$. Modify the algorithm or add a procedure that will check if there is another leaf node $k \neq j$ such that the sum of the key values of all nodes on the path from r to k, denoted by $sum(r, k) \ = \ sum(r, j)$.
14. Consider a square $S \in [0,1]^2$, that is a square with corner points (0,0) (01),(1,0) and (1,1). Le C be a collection of rectangular tiles, where the area of each tile lies between 0 and 1. The problem is to fit the tiles into the square such that the total area occupied by the tiles is maximum. Write an algorithm that will achieve this objective. Note that the above problem has no polynomial time algorithm till date and has neither proven to be Intractable. So It is suggested that you do not try to write an algorithm to find the

optimum solution. Your strategy should be to maximize the area, which may not be the maximum/optimum answer.

15. Consider the following code which performs a derangement of n integers in a given array. A Derangement is a permutation of the n elements, such that no element appears in its original position. For example, a derangement of {0, 1, 2, 3} is {2, 3, 1, 0}.

```
intcountDer(intn)
{
  // Base cases
  if(n == 1)   return0;
  if(n == 0)   return1;
  if(n == 2)   return1;

  // countDer(n) = (n-1)[countDer(n-1) + der(n-2)]
  return(n-1)*(countDer(n-1) + countDer(n-2));
}
```

It is evident that the time complexity of the above recursive code is $T(n) = T(n-1) + T(n-2)$ which is exponential. Modify the code so that the problem can be solved in polynomial time.

16. Consider an array A[1…n] constructed by the following process: we start with n distinct elements, sort them, and then rotate the array k steps to the right. For example, we might start with the sorted array [1; 4; 5; 9; 10], and rotate it right by k = 3 steps to get [5; 9; 10; 1; 4]. Give an O(log n)-time algorithm that finds and returns the position of a given element x in array A, or returns None if x is not in A. Your algorithm is given the array A[1…n] but does not know k.

17. While exploring an ancient temple, Prof. Jones comes across a locked door. In front of this door are two pedestals, and n blocks each labeled with its positive integer weight. The sum of the weights of the blocks is W. In order to open the door, Prof. Jones needs to put every block on one of the two pedestals. However, if the difference in the sum of the weights of the blocks on each pedestal is too large, the door will not open. Devise an algorithm that Prof. Jones can use to divide the blocks into two piles whose total weights are as close as possible. To avoid a boulder rolling quickly toward him, Prof. Jones needs your algorithm to run in polynomial time. Hint: You might want to keep the total weight of each pile to be as close to $W/_2$ as possible.

18. Determine the average processing time T(n) of the recursive algorithm:

```
int myTest( int n ) {
if ( n <= 0 ) return 0;
 else {
 int i = random( n - 1 );
 return myTest( i ) + myTest( n - 1 - i );
      }
 }
```

providing the algorithm random(int n) spends one time unit to return a random integer value
uniformly distributed in the range [0, n] whereas all other instructions spend a negligibly small time (e.g., T(0) = 0).

19. A tenant upon renting a house decides to place his collections of portraits on a wall. To his utter disappointment, he sees that the wall has a lot of nails, of which he can use some of the nails and some nails are not required. He has to now decide which nails have to be removed and which should not be removed. The only information he has with him is that each portrait covers an area of at least 2 sq. ft. Is there an efficient algorithm to decide which of the nails must be removed and which need to be left as it is, so that he can hang his portraits without having any two portraits overlap each other?