INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE – 247 667
Data Structures (CS 102), B.Tech CSE and ECE

Assignment: 2 Spring Semester 2015-16

---

1. An array A[0, ... 2n] is ***wiggly*** if A[0] $\leq$ A[1]$\geq$ A[2] $\leq$ A[3]$\geq$ . . .$\leq$ A[2n-1] $\geq$ A[2*n*]. Given an unsorted array B[0 ... 2*n*] of real numbers, Write an efficient, $\Theta(n)$ time complexity, C++/JAVA program using arrays that outputs a permutation A[0 ... 2n] of B such that A is a ***wiggly*** array.

2. Given a sorted array of *n* numbers, describe $\Theta(n)$-algorithm that, given another number *x*, determines whether or not there exist two elements in the input array whose sum is exactly *x*. Implement the above algorithm using C++/Java.

3. Consider an array A of n numbers each of which is either 0 or 1. We will refer to such an array as binary. Describe an asymptotically efficient algorithm for sorting a binary array A of size n.
   a. Describe your algorithm.
   b. Write a C++/Java code to implement the same.
   c. Analyze correctly the running time of your algorithm and express the running time in either a Θ or an O notation.
   d. Achieve a running time that is the best possible asymptotically.

4. (a) Write a program to reverse an array in O(*N*) time complexity, where *N* is the number of elements in the array.

   (b) Can you propose an algorithm that doesn't use any extra space.

5. Given an array of size *n*, generate and print all possible combinations of *r* elements in array. For example, if input array is {1, 2, 3, 4} and *r* is 2, then output should be {1, 2}, {1, 3}, {1, 4}, {2, 3}, {2, 4} and {3, 4}.

6. Write an efficient algorithm to count the number of Centered Heptagonal Primes in a given array.

   The centered heptagonal number for n is given by the formula $(7n^2 - 7n + 2)/2$ where n = 0,1,2, ...*A centered heptagonal prime* is a centered heptagonal number that is prime. The first few centered heptagonal primes are 43, 71, 197, 463, 547, 953, 1471, 1933, 2647, 2843, 3697, ...Discuss the time complexity of your algorithm. Implement in C++/JAVA.

7. An array is called ZIG-ZAG if it satisfies all of the following rules:

   - It has odd number of elements
   - All even indexed variables are exactly 1 larger than all odd indexed variables (indexing starts at 0)

   An operation **MOVE()** is defined as:

```
MOVE(A[ ],i,j,k):
  Decrement A[i] by k
  Increment A[j] by k
```

   Given an array A, Write a C++/JAVA program to determine whether it can be made ZIG-ZAG using only **MOVE()** operations. If it can, determine the minimum value of the sum of parameters **k** in all the **MOVE()** operations.

8. Write a program to insert the missing characters in the following list/array:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
D	T	A		S	T	R	C	U	R	E	S		I	T		R	O	K	E

9. Determine the space complexity of the following code segment. Justify your answer.

```
sum=0;

    for(i=1; i<=n; i++)
        for(j=1; j<=n; j*=4)
            sum+=1;
        int *arr;
        arr=new int[sum];
```

10. Write an efficient algorithm to count the number of words in a given string. Use C++ to implement the algorithm.