1. a) Consider the following program outline where |A| represents the number of items in array A.

```
method Review (array A) {
    if (|A| > 1) {
Do something to A that takes time O(|A| log(|A|));
Split A into two equal size pieces called B and C;
Review(B); Review(C);
Modify A using O(|A|) time;
} }
```

What is the recurrence that describes time taken by this algorithm in terms of master theorem and reason for the same?

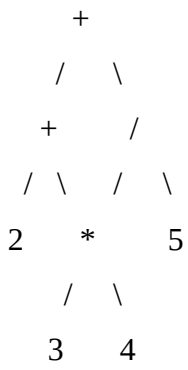
b) A draft report has five chapters. The table shows the lengths of the chapters & their importance where the scale is from 1(low) to 10(high). The report must be at most 600 pages long. The problem is to edit the report so that the overall importance is maximized. Implement the fractional knapsack algorithm using Greedy Programming.

Chapter	Pages	Importance
1	120	5
2	150	5
3	200	4
4	150	8
5	140	3

c) Suppose you have inherited the rights to 500 previously unreleased songs. You plan to release a set of five compact disks with a selection of these songs. Each disk can hold a maximum of 60 minutes of music, and a song can't overlap from one disk to another. Since you are a classic music fan and have no way of judging the artistic merits of these songs, you decide the following criteria a)The songs will be recorded on the set of disks in order by the date they were written b) The number of songs included will be maximized. You have a list of the length of the songs, L1,L2,….L500 in order by the date they were written. (Each song is less than 60 minutes long). Give an algorithm to determine maximum number of songs that can be included in the set satisfying the given criteria.

d) A group of n students takes the theory & practical sections of the ADS. Nobody is tied on either a theory score or on a practical lab score. Let's call a student a winner if nobody beat him/her on both sections of the exam. The losers are students that were beaten by somebody on both sections of the exam. Come up with

	an efficient algorithm for identifying winner. Then briefly summarize the main idea so that it is to be solved considerably faster than $O(n^2)$. e) Write a note on Fibonacci heap. How we merge two Fibonacci heaps. f) Given a file containing a large number of distinct 30-bit binary words. What is a good way to find all complementary pairs that are present? Two words are complementary when one has 0 wherever the other has 1, and conversely; thus they are complementary iff their sum is $(11...)_2$ when they are treated as binary numbers.
2	a) Give a recursive procedure for swapping left and right children of every node in a binary tree so that after conversion it will be a mirror image of the original tree. b) Write an algorithm for a function changepriority(q, x, v) to change the priority of a particular element from x to v in a priority queue q , so that after the change the queue remains the priority queue. c) You are given a k×k checkerboard with a nonnegative number in each square (the square colors are not significant). A token is moved from square to square on the board. Each time the token enters a square it is charged the amount written in that square. Assume that the only legal moves are to the right, down, and diagonally right down. Give an algorithm that runs in $O(k^2)$ time to find cost of the minimum cost sequence of moves beginning in upper left corner and ending at lower right corner. d) Write a method which takes two stack objects as parameters and which returns a new stack object which contains the contents of the first stack, in the same order that they appear in the original, followed by the contents of the second stack in the same order that they appear in the original. Your algorithm should leave two original stacks unchanged when the method completes. e) What you understand by B-Trees. Take an example and show working of basic operations on B-Trees. f) Modify the binary search algorithm so that it splits the input not into two sets of almost equal sizes but into two sets of sizes approximately one third and two thirds.
3	a) Suppose you have an array of 1000 records in which only a few are out of order and they are not very far from their correct positions. Which sorting algorithm will you use to put the whole array in order? Justify. b) The transitive closure of a digraph with n vertices can be defined as an n by n boolean matrix T such that

	T[i,j] =1 if there is a path between ith and jth vertex, otherwise it is zero. Design an algorithm for computing the transitive closure & give complexity. c) One Popular cryptographic method for encoding a message is substituting another letter for each letter of the alphabet by some formula. We can uniquely decipher a cipher text by reading the substitution backwards. Write a program that takes as an input a plain text generates a substitution and then encrypts plaintext using that substitution. d) Suppose an arithmetic expression is given as Directed acyclic Graph with common sub expressions removed. For example 2+3*4+5/(3*4) would be given as shown. Give an algorithm for evaluating such an expression in time O(n).  e) Define AVL trees. What are the various rotation operations involved. f) Distribute the integers in a list according to their sizes. Creating an array of lists, named bucket. The array bucket has 10 entries. List elements in the range 0 through 99 should go into the list bucket [0], elements in the range 100 through 199 should go into the list bucket [1] and so on, and all elements that are 900 or over should go into the list bucket [9]. Assume your procedure takes two parameters only, the list of elements to distribute and the array bucket.
4.	a) The input to the problem consists of an ordered list of n words. The length of the ith word is w_i, that is the ith word takes up w_i spaces. (For simplicity assume that there are no spaces between words.) The goal is to break this ordered list of words into lines, this is called a layout. Note that you cannot reorder the words. The length of a line is the sum of the lengths of the words on that line. The ideal line length is L. No line may be longer than L, although it may be shorter. The penalty for having a line of length K is L-K. The problem is to find a layout that minimizes the total penalty. Prove or disprove that the following greedy algorithm correctly solves this problem for a and/or b.

For i= 1 to n

Place the ith word on the current line if it fits

else place the ith word on a new line

a) *The total penalty is the sum of the line penalties.* b) *The total penalty is the maximum of the line penalties.*

b) Using Kruskal's method of MST find the five edges

	A	B	C	D	E	F
A		7	6	1	3	12
B			4	5	8	10
C				9	11	15
D					2	14
E						13
F						

Show the edges in chronological order: Edge 1 Edge 2 Edge 3 Edge 4 Edge 5

c) Consider a sequence of n distinct integers. Design and analyze a dynamic programming algorithm to find the length of the longest increasing subsequence. For example consider the sequence 45 23 9 3 99 108 76 12 77 16 18 4

The longest increasing subsequence is 3 12 16 18 having length 4.

d) Discuss Knuth Morris Pratt Algorithm for string comparison.

e) A graph is said to be bi-partite if all its vertices can be partitioned into two disjoint subsets X and Y so that every edge connects a vertex in X with a vertex in Y. It is also called as two-colorable graph where every edge has its two vertices colored in different colors. Design a DFS or BFS based algorithm to see whether a graph is bipartite or not.

f) Devise an algorithm based on Horner's Rule for converting a string of alphanumeric digits to its numeric value. If there is an alphabet in between, return it as a bad string and exit the program. Assume a function ConvertDigit exists for converting an alphanumeric digit to its integer equivalent e.g. ConvertDigit ('5') = 5. Also assume that the alphanumeric digits are input one at a time in an online fashion, so that the total number of digits is not known in advance.