1	Answer the following a) Discuss Bubble sort, Selection sort and Insertion Sort in terms of number of comparisons required to do the sorting and which one will be relatively better. b) Discuss the use of comparison and non comparison sorts. How we will decide whether to use radix sort and in which cases it will become unwise to be used. c) Amortized complexity and Expected complexity are becoming important as compared to Worst case complexity. Discuss the reasons. d) In the cases where both Greedy and Dynamic can be used for getting the optimal results, which one will be preferred and why?
2	Write complete algorithms for the following problems a) KMP for string matching b) Bellman Ford Algorithm for shortest distance c) Fractional Knapsack algorithm d) Integer Multiplication using Divide and conquer strategy
3	a) You are planning a cross-country trip along a straight highway with n+1 gas stations. You start at gas station 0, and the distance to gas station i is d_i miles ($0 = d_0 < d_1 < . . . < d_n$). Your car's gas tank holds G > 0 gallons and your car travels m > 0 miles on each gallon of gas. You start with an empty tank of gas (at gas station 0), and your destination is gas station n. You may not run out of gas, although you may arrive at a gas station with an empty tank. As you are very rich, rather than trying to plan the cheapest trip, you want to minimize the total number of stops you need to make (you stop at a gas station only if you need to buy gas there). Assume that G, m, and all d_i are positive integers (except $d_0 = 0$) and are polynomially bounded as a function of n. Assume all d_i are distinct. If possible, give a polynomial time algorithm to determine the value of the optimal solution (i.e. the minimum number of stops). b) You have a sequence S of n characters from an alphabet of size k <= n; each character may occur many times in the sequence. You want to find the longest subsequence of S where all occurrences of the same character are together in one place; for example, if S = aaaccaaaccbccbbbab, then the longest such subsequence is aaaaaaccccbbbb = aaa__aaacc_ccbbb_b. In other words, any alphabet character that appears in S may only appear in one contiguous block in the subsequence. If possible, give a polynomial time algorithm to determine the value of the optimal solution (i.e. the length of longest such subsequence).
4	A boolean chain is a sequence of n boolean random variables $X_1, . . . ,X_n$ such that the value of X_i depends only on the value of X_{i-1}. Graphically we can represent the chain as shown: X1 –> X2 –> . . . -> Xn Each arrow represents a probabilistic dependence of X_i on X_{i-1}; this dependence consists of four probabilities: • $\Pr[X_i = T \mid X_{i-1} = T]$ • $\Pr[X_i = T \mid X_{i-1} = F]$ • $\Pr[X_i = F \mid X_{i-1} = T]$ • $\Pr[X_i = F \mid X_{i-1} = F]$ For the first node in the chain we simply have $\Pr[X_1 = T]$ and $\Pr[X_1 = F]$. Notice that the probability of any assignment ($X_1 = x_1, X_2 = x_2, . . . , X_n = x_n$), where each xi is either T or F, is therefore equal to $\Pr[X_1 = x_1] \cdot \Pr[X_2 = x_2 \mid X_1 = x_1] \cdot . . . \cdot \Pr[X_n = x_n \mid X_{n-1} = x_{n-1}]$ (a) Give the basic constraints on these probabilities, beyond the fact that all probabilities are between 0 and 1. (b) Consider the problem of finding the most likely truth assignment to the variables $X_1, . . . ,X_n$. Formulate this problem as a single-source shortest path problem in some graph G. Be explicit about what the nodes and edges of G are, and what the weights on these edges should be. What algorithm would you use to find the shortest path and what is its running time as a function of n? c) Give a dynamic programming algorithm to find the most likely assignment in time O(n). Hint: It may be helpful to introduce an auxilliary variable X_0 which is always true, so that X_1 may be treated almost uniformly with $X_2, . . . ,X_n$.
5	Write notes on the following a) NP-Complete and NP problems along with the diagram and examples b) Matrix multiplication algorithm using parallel programming c) Disjoint Set data structure along with its improvements d) B-Tree data structure along with its applications and benefits