1. XYZ Ltd. decides to take all its employees to a picnic. It is decided that they play a game of Ivri. This game needs two teams of equal numbers. XYZ has N employees organized like a tree structure. Each employee (except CEO, who is at root) has an immediate superior (a parent node). Each employee has a preference value $P_i$ with which he wishes to play against his immediate superior. An employee’s preference value to play against anybody else is zero.

The task is to divide the N employees into two groups, while maximizing the total preference values. When an employee (with preference value $P_i$) and his/her immediate superior are selected into opposite teams, the total preference value is increased by $P_i$.

Give a dynamic programming algorithm for this problem. Give the running time and pseudocode. Make assumptions as required while calculating the running time. Please list your assumptions along with the analysis. (20 Points)

2. Let G(V,E) be any connected undirected graph. A bridge of G is defined to be an edge of G which when removed from G, will make it disconnected. Present an $O(|E|)$ time algorithm to find all the bridges of G. (10 points)

3. It is easy to see that for any graph G, both DFS and BFS will take almost the same amount of time. However, the space requirement may be considerably different. (10 points)

(a) Give an example of an n-vertex graph for which the depth of recursion of DFS starting from a particular vertex v is n-1 whereas the queue of BFS has at most one vertex at any given time if BFS is started from the same vertex v.

(b) Give an example of an n-vertex graph for which the queue of BFS has n-1 vertices at one time whereas the depth of recursion of DFS is at most one. Both searches are started from the same vertex.

4. A game of SWAP works as follows: you are given a two dimensional table $S[i, n]$ of real numbers between 0 and 2. Initially you start with an initial position $i$ and initial value $c$. According to the rules, you can swap your initial position with any other position between 1 and $n$. After each swap from $i$ to $j$, your start value is multiplied by $S[i,j]$. The player who (atleast) doubles his initial value and end at his initial position wins the game.
You have realized that given the swap table, you can find a series of swaps whose product is greater than 1 and end at the same start position. If you find such steps you can complete the game by repeating them till necessary. There is a possibility that some $S$ tables does not have such series of swaps. Your task is to find weather a given $S$ table is solvable or not.

Design a dynamic programming algorithm to solve this task. You must give the running time and prove that your algorithm is correct. (15 points)

5. Pearls have been prized for their beauty and rarity for more than four thousand years. Natural pearls are collected in the coastal regions of Asia, Africa and America. Since pearl collection is dangerous, researchers are asked to build a robot that can do this task. A task for the robot consists of $n$ potential pearl sites all located along a single coastal line. Robot starts at site 1, but can finish at any site. To make things easier we assume that all the sites are on a straight line, one after the other. The time taken to travel from site $i$ to site $i+1$ is $t_{i,i+1}$ hours, $1 \leq i \leq n-1$. Assume that robot’s battery lasts for $m$ hours. Also, the expected number of pearls per hour is $p_i$ at site $i$, $1 \leq i \leq n$.

It was noticed that each hour of pearl collection decreases the expected number of pearls to be found in the next one hour, by a constant amount. This amount varies from site to site and is given as $d_i$ for site $i$, $1 \leq i \leq n$.

Design a dynamic programming algorithm to maximize the total pearls found. Give the pseudo-code and run time complexity. (15 points)

6. The problem of OBST (Optimal Binary Search Tree) is as follows: You are given a set of identifiers along with their search probabilities. The task is to form a binary search tree, which minimizes the expected number of comparisons. To be more precise: Let the given identifiers $a_1, ..., a_n$ be ordered ascendingly, let their given search probabilities be $p_1, ..., p_n$. Also let $q_0, ..., q_{n-1}$ be the given, where $q_i$ is the probability that an identifier not in the tree but between $a_i$ and $a_{i+1}$ ($q_0$ is the probability that the identifier being search for is less than $a_1$). Design a dynamic programming solution that produces the optimal binary search tree given these inputs. (15 points)

7. Ravi is planning to drive from Gainesville to Portland. He wants to use rental cars (one or more) for the trip. He visits multiple cities $c_1, c_2, ..., c_n$ in order ($c_1$ is Gainesville and $c_n$ is Portland). If a car is picked up in $c_i$ and dropped off in $c_j$, $j > i$, you pay $C(i,j)$ dollars. Note that the rental costs could be non-monotonic. For example, it might be that $C(2,3)$ is 100, but $C(2,4)$ is 75. Give an algorithm to choose a set of car (rentals) that minimize the cost of the trip. Note that you can never be without a car and you can rent only one car at a time. (15 points)