**Design and analysis of algorithms**

**Project Report**

|  |
| --- |
| KASHFA FAROOQ 19i-0421 (Question-1)  AMNA SHAFIQ 19i-1978 (Question-2)  ABEEHA FATIMA 19i-0742 (Question-3) |

**Question-1**

This problem uses hill climbing search technique to find possibly optimal truth assignments for variables in Boolean formulas. It is a local search algorithm which continuously moves in the direction of increasing elevation/value to find the peak of the mountain or best solution to the problem. It terminates when it reaches a peak value where no neighbor has a higher value. The formulas are in conjunctive normal form (ANDs of ORs). The fitness of an assignment is the number of clauses (ORs) that the assignment satisfies. I am using a vector to represent the formula where each element is a vector of integers representing the clause, and each integer represents the variable index.

In hill climbing algorithm, we have to check if the random assignment satisfies the formula. If not, flip the bit with the highest fitness, that is the bit that appears in most clauses. If we reach a plateau, in which all the bits yield the same or a lower fit fitness, choose a new random assignment.

There are 33 functions with n complexity and 2 with n2 complexity therefore the total complexity of this solution is 33n+2n2

**Question-2**

In this question I have used vectors to store the data for both prices and sizes of land.

Part-1 of this question is done by simple recursion without memorization algorithm and the time complexity of this function is O(n2).

Part-2 of this question is done by recursion with memorization algorithm and the time complexity of this function is O(n2).

Part-3 of this question is done by iterative method with memorization algorithm and the time complexity of this function is O(n2).

Time complexity of the whole program is 3O(n2) which can be assumed as O(n2).

**Question-3**

In this question I have used character arrays to store the data for the text and the pattern.

I have also used a class **Position** having x and y data members as coordinates of the array to store the starting positions of the pattern found in the text array.

Time complexity of the program is:

-> O(n/m)\*n\*m\*m + O(nlogn)

-> m2 \* m +nlogn

-> O(mn2)