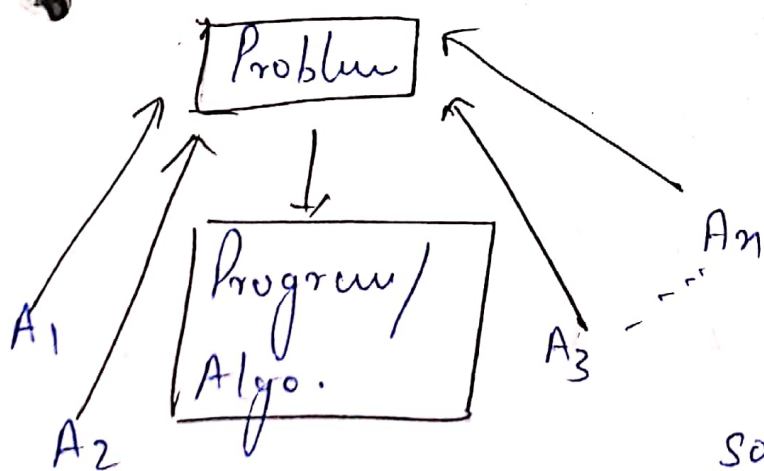


Algorithms - Algorithm is a finite set of steps to solve a particular problem. An algo. must have the following properties -

- i) Finiteness: must complete after finite no. of steps
- ii) Absence of Ambiguity: Each steps must be clearly defined.
- iii) Input/Output: Number and types of input and output must be specified.
- iv) Feasibility: It must be possible to perform each instruction.
- v) Termination: Must be terminated after finite no. of steps.



If we have a problem then we are going to write a program or we can write an algo to solve that problem. But for solving a problem we can have more than one program or more than one algo. Then we have to choose an algo which is best suitable.

Algorithm Analysis:-

Algorithm analysis is an important part of computational complexity theory, which provides theoretical estimates for the resources needed by any algorithm which solve a given problem.

Algorithm Classification OR

Algorithm Design Techniques:-

For any problem there may be many ways to design algo-

- | | |
|-------------------------------|------------------------|
| i) Recursive Algo | ii) Backtracking Algo. |
| iii) Divide and Conquer Algo. | iv) Greedy Algo. |
| v) Branch and Bound | vi) Brute Force Algo |
| vii) Dynamic Programming | viii) Randomized Algo. |

i) Recursive Algo: In this type of algo the function call itself to solve the problem.

ii) Backtracking: It try each possibility untill they find the right soln. It is D.F.S. based algo.

iii) Divide and Conquer Algo:- Divide the original problem into subproblems. Solve each subproblem individually and then combine all the solutions.

iv) Greedy Approach:- In this approach we optimize a function by making ^{have diff} choices which are the best locally but do not look at the global problem. The solⁿ is good but not necessarily the best one.

v) Branch and Bound:- Generally used for optimization problem. As the problem progresses a tree of subproblem is formed.

vi) Brute Force Algo:- It simply tries all possibilities until a satisfactory solⁿ is found.

vii) Dynamic programming:- It remembers past results, storing them and uses them to find new results.

viii) Randomized Algo:- (i) uses a random no. at least once during the computation to make a decision. Ex. In Quick sort using a random no. to choose a pivot.

Complexity of Algorithms:-

- i) Time Complexity:- Running time of the program as a function of the size of the input.
- ii) Space Complexity:- Space complexity means that how much space an algo. needs to complete the task.

Best, Worst and Average Case Complexity:-

- * The best case complexity of the algo. is the function defined by the minimum no. of steps taken on any instance of size n .
- * The avg. case complexity of the algo. is the function defined by the avg. no. of steps taken on any instance of size n .
- * The worst case complexity of the algo is the function defined by the max. no. of steps taken on any instance of size n .