NAME :- KRUNAL RANK.

ROLL NO:- U18CO081

CLASS:- B TECH 3RD YEAR,
COMPUTER ENGINEERING.

U18CO081 26/11/2020

DAA
Tutorial 9

Ans 1:
The characteristics of greedy algorithm are as follows:-
- The best option is chosen in the hope that it will lead to an optimal solution.
- There is no certainity that it will lead to an optimal solution.
- The solution is easy to formulate and implement.
- It is more memory efficient.

Ans 2: Let us take the example of Dijsktra's algorithm.
It is an example of a greedy algorithm.
The problem dictates us to find the shortest path between two nodes,
The problem is solved greedily in the following manner:-
- We start with the source node.
- For all its neighbour nodes, we mark them with the minimum distance through which we can visit the neighbouring node. If
  dist[src] + edge weight < dist[dest],
  then we rewrite rewrite destination distance and push it in the priority queue.
- After we process the source node, we then process the node that is in the priority queue and has the minimum distance value.
- Since, the approach has some parameter as a benchmark for choosing the next node, it is a greedy algorithm.

It is different from Dynamic Programming as there is no formulation of states and no transition is involved.

Ans 3: Fractional knapsack problem can be depicted as follows:

Consider a knapsack whose maximum capacity is W. The knapsack needs to be filled with items whose total weights $w_1, w_2, w_3 \ldots w_i$ as well as its value $v_1, v_2, v_3 \ldots, v_i$ is given. Now, the item can be partially filled as suggested by the name "fractional".
The goal is to maximize the total value of the items used to fill the knapsack such that the total weight of items do not exceed the capacity of knapsack.

It is clearly a greedy algorithm. The algorithm is as follows:

- First find the value per weight ratio of all items, say $vw_1, vw_2, \ldots v_i$
- Sort all the items according to the value per weight ratio in descending order.
- Initialise ans = 0, totalWeight = 0
- Repeat the steps until all items are processed or total weight exceeds the knapsack capacity.
  → If totalWeight + w[i] <= W then, ans = ans + v[i];
  totalWeight = totalWeight + w[i]
  → Else,
  usedWeight = W - totalWeight
  ans = ans + usedWeight * vw[i]
  break
  i = i+1
- Return ans.

Ans 4: Yes, selection sort is a greedy algorithm.
The function that involves a greedy ~~proc~~ procedure is ~~to~~ the "find" function that finds the next possible value, Suppose, for sorting in ascending order, it is the next minimum value.

Ans 5: The solution needs to be constructed using ~~the~~ a greedy approach :-

- Sort the intervals in the increasing order of their finishing time.
- Iterate through all given intervals while maintaining a list of acceptable tasks.
- If a current task does not affect any task in the list, then add this interval to the list.
- The complexity of the algorithm is O(NlogN).