CS & IT ENGINEERING

Algorithms

Greedy Method

Lecture No.- 01

By- Dr. Khaleel Khan sir

Recap of Previous Lecture

Topic Divide and Conquer

Topics to be Covered

Topic

Introduction to Greedy Method

Control Abstraction

Knapsack Problem

GREEDY METHOD:

→ Used for Solving Problems, whose Solutions are viewed as a result of making a Set/Sequence of decisions;

→ These decisions are made in a Step-wise manner;

→ At each Step out of all options, Greedily Select that option, which Satisfies the given criteria of the problem;

Terminology:

→ Problem Definition,

→ Constraints (Conditions / Requirements)

- Implicit → Row/Column/Diag (Local)
- Explicit (Boundary)

$$\boxed{1 \le x[i] \le n}$$

→ Solution Space:

" All possible ways of organizing inputs, Satisfying only explicit Constraints "

$\langle 1, 2, 3, 4 \rangle$ ×

① n-Queens:

n-queens $(q_1 \dots q_n)$

$n = 4$; $(q_1 \dots q_4)$

	1	2	3	4	
q_1		•			$x[1]$
q_2	×	×	×	•	$x[2]$
q_3	•				$x[3]$
q_4	×	×	•		$x[4]$

int $x[1..n]$, $x: \langle 2, 4, 1, 3 \rangle$
$x: \langle 3, 1, 4, 2 \rangle$

$x[i]$ = pos (column) in which Q_i is placed

$n = 4$

$\times\ [1, 2, 3, 4]$ ✓

$[1, 2, 4, 3]$

$[1, 3, 2, 4]$

$[2, 4, 1, 3]$

$\vdots$

$\Big\}\ \underline{\underline{n!}}$

$n = 3$

$$\left.\begin{matrix} 1, 2, 3 \\ 1, 3, 2 \\ 2, 1, 3 \\ 2, 3, 1 \\ 3, \cancel{1}, 2 \\ 3, 2, 1 \end{matrix}\right\}$$

→ Each problem will have its Solution space as a fn of Input Size

Ex: for n-Q's Problem, it is $n!$

: 2^n ; ∞ ; n^2 ; n^n

→ **Feasible Solutions:** are those Solutions in the Solution Space that Satisfy implicit Constraints of the Problem,

Problems may have multiple Feasible Solutions,

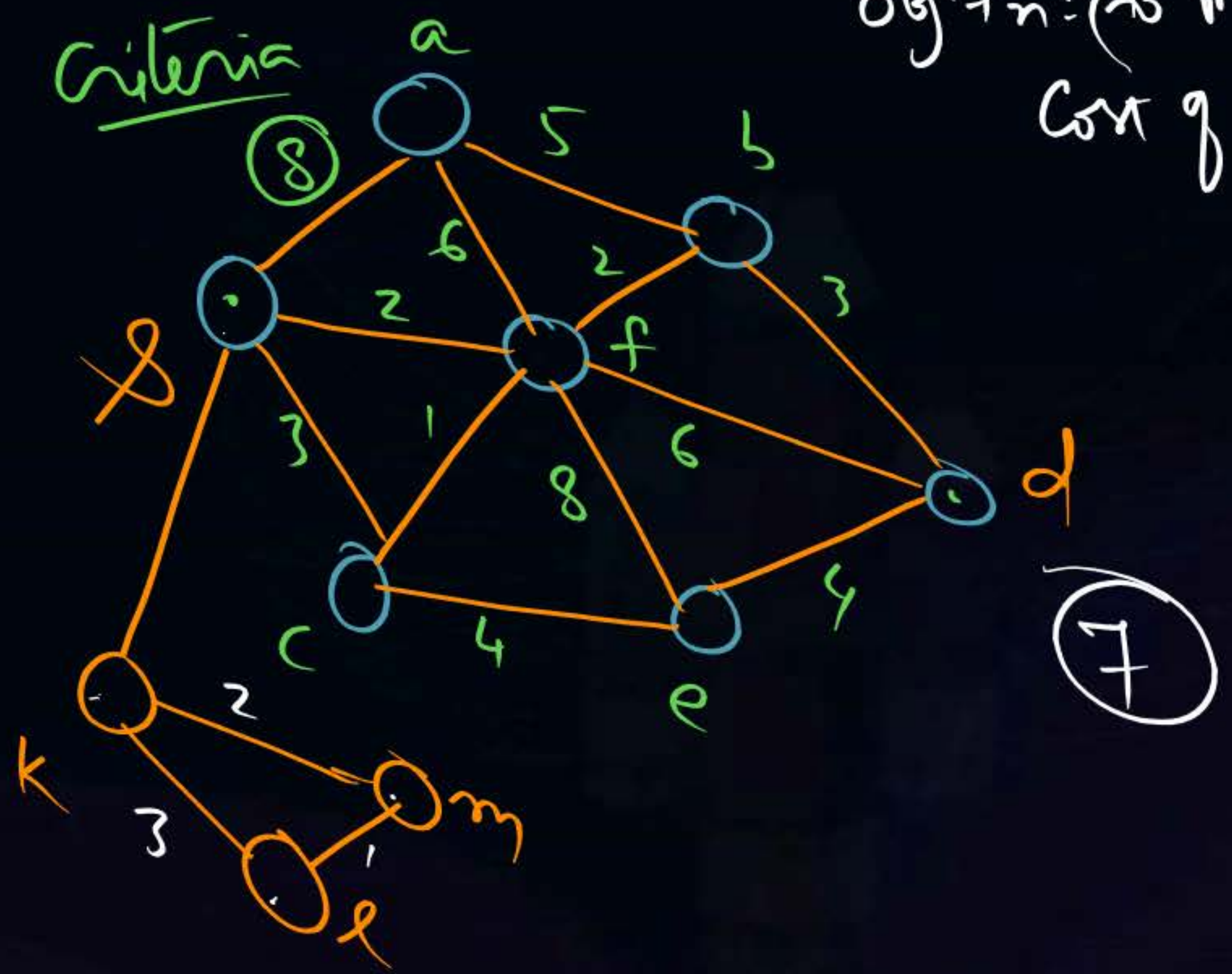
→ **Objective function:**
(Few problems may have Objective fn, which tries to Min./Max. a given Criterion of the Problem)

→ **Optimal Solution:** is that feasible Solution, which Satisfies the objective fn;

En: Shortest Path

Obj. fn: (to Minimize Cost of Path)

Criteria



→ Optimal Solution always refers to the value and hence it is always unique,

Feasible Soln

Q) What is objective fn of N-Queens Problem?

→ No objective fn

Problem (P)

Searching

n-queens

Sorting

(Shortest Paths)

Graph Coloring

Decision

→ Its result is always either Y/N (T/F) or (0/1)

(Feasible Solns)

Optimization

→ Requires to determine a Max/Min value of a given criterion,

(optimal Soln (objective Fn))

1) CPU Scheduling → Optimization

2) Page replacement → Optimization

3) Disk Scheduling → Optimization

4) Banker's Safety Algo → Decision

5) GATE 2024 Problem

6) Congestion Control → Optimization

GCSIT

GDA

Topic : Greedy Method

Control Abstraction

General Method

```
Algorithm Greedy(a, n)
    // a[1 : n] contains the n inputs.
{
    solution := ∅; // Initialize the solution.
    for i := 1 to n do
    {
        x := Select(a);
        if Feasible(solution, x) then
            solution := Union(solution, x);
    }
    return solution ;
}
```

1 . solution := ∅

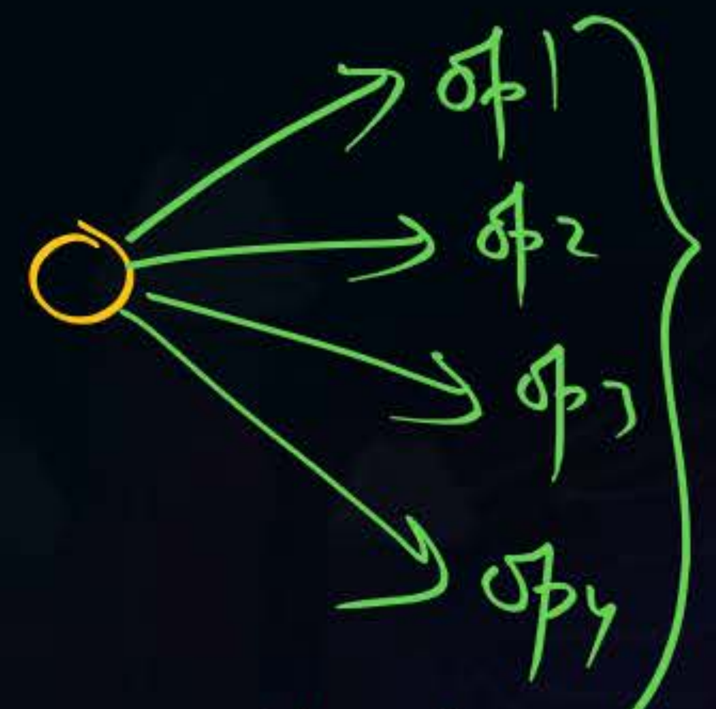
2 . for i := 1 to n do

Select(a) → O(1)

→ O(log n)   O(n)

Union — ADD → O(1)

(Principle of local Optimality)



Time complexity of atleast O(n)

1. KNAPSACK Problem (KNAP)

→ Given a KNAPSACK (Bag) of Capacity 'M'

→ Given 'n'-objects ($O_1, O_2 \ldots O_n$)

✓ Weight (w_i)   Profit (P_i)   Decision (x_i) ?

(Max. Profit)

→ If object of weight (w_i) is put into the KNAP, then KNAP gets filled up by wt. w_i & we get profit 'P_i';

→ Maximize the Profit Subject to the Condition that the total wt. being put into the KNAP does not exceed its Capacity.

KNAPSACK Problem

Fractional

Real

Continuous

$$\boxed{0 \le x_i \le 1}$$

$\langle \infty \rangle$

$x_i = 0/1$

Binary (0/1)

⟨ either include it totally or exclude it ⟩

1) $\sum_{i=1}^{n} w_i \leq M \Longrightarrow$ Total Profit $= \sum_{i=1}^{n} P_i$

$\hookrightarrow x_i = 1$

<No-decision>

<Trivial>

2) For solving the Problem, then

$$\boxed{\sum_{i=1}^{n} w_i > M}$$

Explicit Constraint

$x_1 = ?$

$x_2 = ?$

$x_3 = ?$

$\vdots$

(Decision on O_i) x_i — $w_i * x_i$ / $P_i * x_i$

$$\text{Max} \sum_{i=1}^{n} P_i * x_i \longrightarrow \text{object. fn}$$

$$\text{Subject to} \sum_{i=1}^{n} W_i * x_i \le M \longrightarrow \text{Impl. constn't}$$

$$\text{Where } 0 \le x_i \le 1 \Longrightarrow \text{Expl. constraint}$$

Soln Space

(G.M) Fractional "∞"

(D.P) 0|1 KNAP "2^n"

$\langle x_1, x_2 \ldots x_n \rangle$ $\{ \infty \}$

$x_i = 0|1$

$\langle x_1, x_2 - - x_n \rangle$ $\{ 2^n \}$

① $n = 3;\ M = 20;\ \langle w_1; w_2; w_3 \rangle = \langle 18; 15; 10 \rangle$

$\langle P_1; P_2; P_3 \rangle = \langle 25, 24, 15 \rangle$

o_1

(Fractional KNAP)

$\langle x_1\ x_2\ x_3 \rangle$

a) Greedy about Profit:

$x_1 = 1$

$x_2 = 2/15$

$x_3 = 0$

$\sum w_i x_i = 20$

$\sum P_i \cdot x_i = \left(25 + \frac{2}{15} \times 24 + 0\right)$

$= \boxed{28.3}$

b) Greedy about weight:

$x_3 = 1$

$x_2 = 10/15 = 2/3$

$x_1 = 0$

$\sum w_i x_i = 20$

$\boxed{\sum P_i \cdot x_i = 31}$

$w_i \longrightarrow P_i$

$1 \longrightarrow ?$

(P_i / w_i)

1) $T(n) = 2T(\sqrt{n}) + \dfrac{\log n}{\log\log n} \quad , n > 4$

$\quad = c \quad , n \le 4$

2) $T(n) = 2T(n/2) + \dfrac{n}{\log n} \quad , n > 1$

$\quad = c \quad , n = 1$

THANK - YOU