TSP

$$m[S, c_i] = \min \left\{ \begin{array}{l} m[S \setminus \{c_i\}, c_j \in S \setminus \{c_i\}] \\ + d(c_j, c_i) \end{array} \right\}$$

c_1 hoche src ; c_i teh shesh
Min. len. er path
S er shob use kore, c_i teh shesh hoy

Longest Path can't have fixed c_1/src.

TSP ⇒ input: given Graph G will be complete.

$d(c_i, c_j)$

$c_1 \Rightarrow$ vertices
$c_2 \Rightarrow$

incomplete edges will have (+INF)

output: cycle ...
cost ... minimum

$O(n! * n)$

TSP

- 'n' length er ekta path banai kore ... loop back.

, Arbitrary vertex fixed source "s"/c_1

, Path len. baraite thakbo.

$$c_1 - \{\overbrace{c_2, c_3, \ldots, c_n}^{\boxed{S}}\}$$

c_1 shuru ; , kai giye shesh hobe ota thi jani na
, loopback korano jabe
, S er shob vertex diye giye,

~~xxxxxx~~

S : c_1 theke shuru kore ... c_i is the last vertex

c_i : c_i lies in S

$$c_1 - \{ \ldots \boxed{S} - c_i \}$$

OPT/m $\Rightarrow$ array
set diye indexing i.e. hashmap

$m[S, c_i]$

c_i er immediate ager vertex, then oita theke c_i teh jabo otar min. len.

$$m[S, c_i] = \min_{\text{all possible } c_j} \left\{ m[S - \{c_i\}, c_j] + d(c_j, c_i) \right.$$

where $c_j \in S \setminus \{c_i\}$

Extra jinish all $(c_i - c_1)$ abar jaite hobe.

(8) $TSP \leq_p LP$ oita diye kora jabe ??

$f(\pi) = c_{\pi_1}, c_{\pi_2}, \ldots, c_{\pi_n}$ ← π : array/list; permutation of 1 to n.

source jani na ; c_i teh shesh

$OPT[S, c_i]$:- "S" er jotogula use kore ; c_i teh giye shesh koro :

$V \Rightarrow$ all vertices set.

$OPT[V, c_i] \Rightarrow$ ~~R~~

Bam dik theke <u>OPT</u>

$OPT[c_i, S]$: c_i theke shuru
{ uses max no. of vertices in S
{ or max total weight of vertices in S

i.e. c_i — S

longest path in S

, c_i belongs in S

c_i starting vertex of S.

NOT necessarily all vertices in S.

- $\text{OPT}[c_i, S] = \max\limits_{\substack{\text{for all} \\ c_j \in S - \{c_i\}}} \left\{ w(c_i, c_j) + \text{OPT}[c_j, S - \{c_i\}] \right\}$

  $c_j \in S - \{c_i\}$

  (c_j is neighbor of c_i)

If $(c_i, c_j) \notin E$, then $w(c_i, c_j) = -\infty$ (-INF)

since maximization problem.

- ~~Final solⁿ~~ $\cancel{\max\limits_{c_i} \{ \text{OPT}[c_i, V] \}}$

- Base case :

  $\text{OPT}[c_i, c_i] = 0$

- Final solⁿ :

$$\max_{c_i \in V} \left\{ \text{OPT}[c_i, V] \right\}$$

Time

$O(n^2 . 2^n) \rightarrow O^*(2^n)$

Space $\Omega(2^n)$

$\because 2^n * 2^n = 2^{2n} = 4^n$

Space: $\Omega(2^n)$

Time: $O^*(2^n)$

---

we want polynomial space $\Omega^*(1)$; but time $O(4^n)$

S ke bhangbo duit equal set -

c_1, c_2, c_3, c_4

c_1, c_2 $\quad$ c_3, c_4

Book : Exact Algorithms

Chapter : 10

$$\sum \binom{n}{k} k! = \sum_k \frac{n!}{(n-k)!}$$