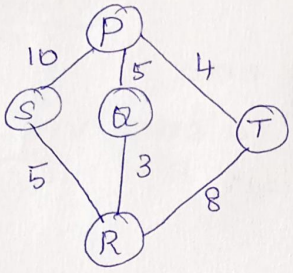
Uniform Cost Search (UCS) Algorithm

→ Uniform cost search is a search algorithm used to find a path to the goal node which has lowest path cost (total cost).

→ This type of algorithm is used when different cost is available for each edge.

Ex :- 1

Let us consider :-



P → Start state
R → Goal state.

Step 1 :-

→ Visited root node 'P' first.

→ from P, 3 edges going are S, Q & T

↳ from these three.

T is having lowest path cost from P to T. So visit T & mark as visited Node.

Visited :-
P, T, R.

→ from T, only one edge is there. & i.e 'R' (also R is a goal state).

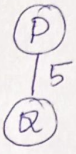
visit 'R'

P → T → R = 12. → ①

→ We need to explore, other edges of P ie S, Q even if we reached to goal state.

Step 2 :-

from P the edges of P to S & Q. Q is having smallest cost. So explore that path.

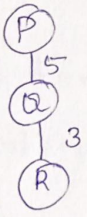


visited
P

visited
P, Q

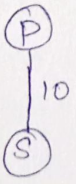
→ From Q, only one edge is there. i.e 'R' (also it is a goal state. Visit R

visited
P, Q, R.



olp sequence :-
P – Q – R : 8 → (2)

Step 3 :- Now explore P ⇒ S.

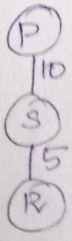


visited
P

visited
P, S.

From S, there is only one edge i.e 'R' also it is a goal state

visited
P, S, R.



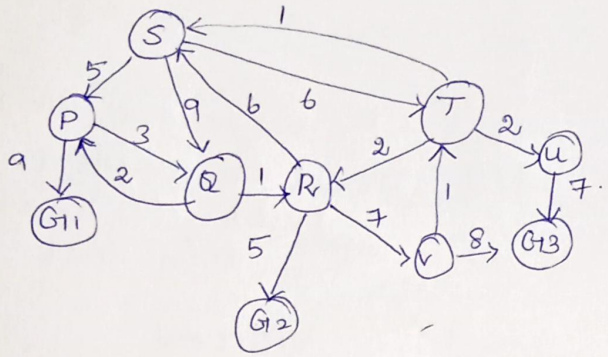
olp seq :- P – S – R : 15 → (3)

From ①, ② & ⑤.

Low cost : ~~co~~ output sequece with total cost is ②

i.e. $P \rightarrow Q \rightarrow R = 8$

Ex:- 2.

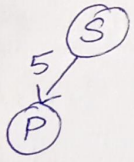


Here :- Goal state : $G_1 = G_2 = G_3$ (same.)

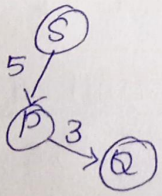
initial state : S

From S outgoing edges are P, Q, T from these three edges. P is having lowest cost.

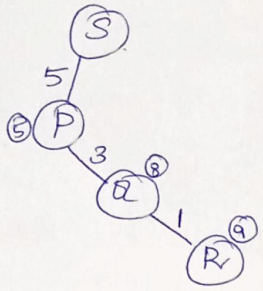
Visited
S, P.



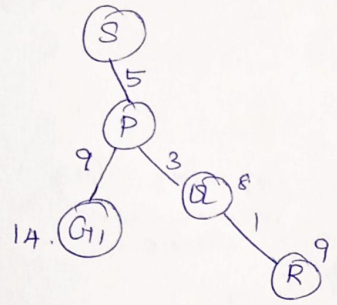
From P outgoing edges are G_1 & Q. Q is having lowest cost.



From Q outgoing edge is R with min cost. 1



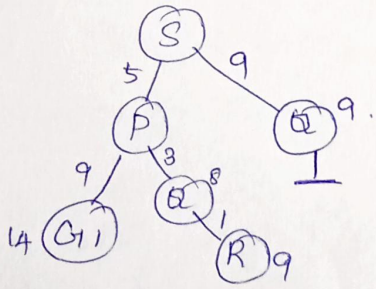
Now explore other P edge i.e G_1. G_1 is a Goal, we reach still need to explore other path to search min cost path.



visited

S, P, Q, R, G_1.

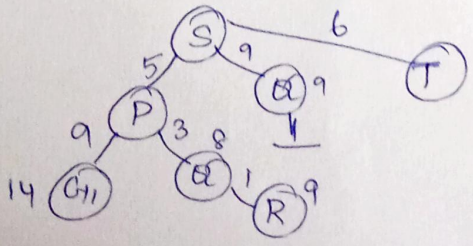
From S, P is explored Now explore Q & T Q is already visited, so don't explore, quit that path



visited

S, P, Q, R, G_1.

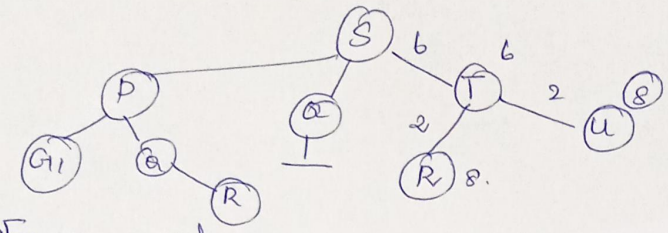
From S, Explore path Q. T



visited :

S, P, Q, Q, G_1, T

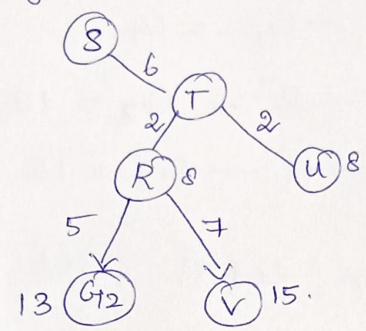
From T outgoing nod is R, & U



Visited
S, P, Q, R, G1, T, R

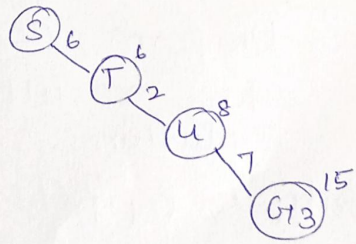
From already we have path Q → R of cost 9. also T to R cost 8, T to U cost 8.

Explore min cost path T to R. from R. there are 2 edge G2 & V



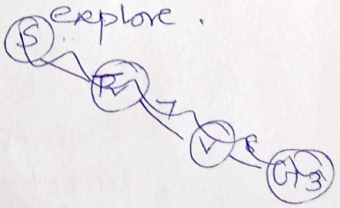
Visited
S, P, Q, R, G1, T, R, G2.

Since we reach G2, still explore other path: Q → R of cost 9 quit 'R' as it was. visited R. From R explore path R — V & T — U. Explore T → U as it has lowest cost of 8.



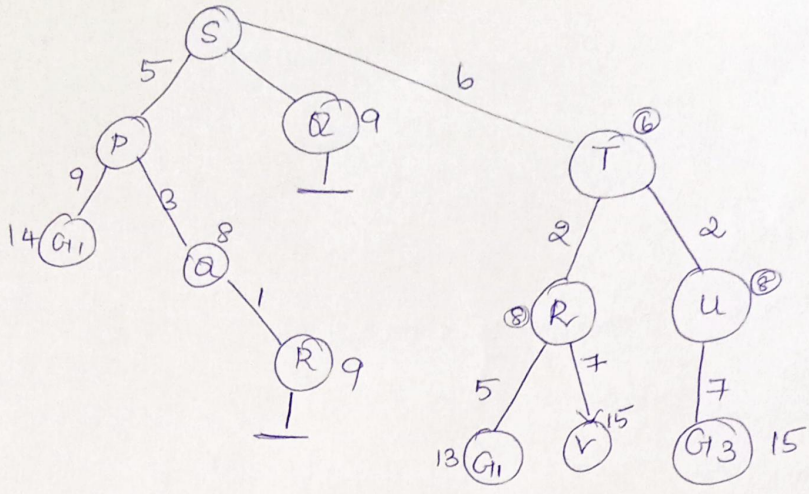
Visited :
S, P, Q, R, G1, T, R, U, G3.

From R — U R — V — G3 cost will be high. So no need to explore.



(6)

visited:
S, P, Q, R, G1, T, R, G1, U, G3.



$S - P - G_1 = 14$

$S - T - R - G_2 = 13.$

$S - T - U - G_3 = 15$

So minimum total cost path is:

$S \rightarrow T \rightarrow R \rightarrow G_2$ with cost = 13.

Performance Evaluation:-

① Completeness: Uniform cost search is complete as it gives solution.

② Optimality :- Uniform cost search is optimal as the result of this is an optimal solution.

③ Time complexity : $O(b^{c/e})$

b → branching factor or Number of nodes.

c → cost of optimal solution.

e → be every action cost at least edge in otherwords "e" is the least cost of an edge.