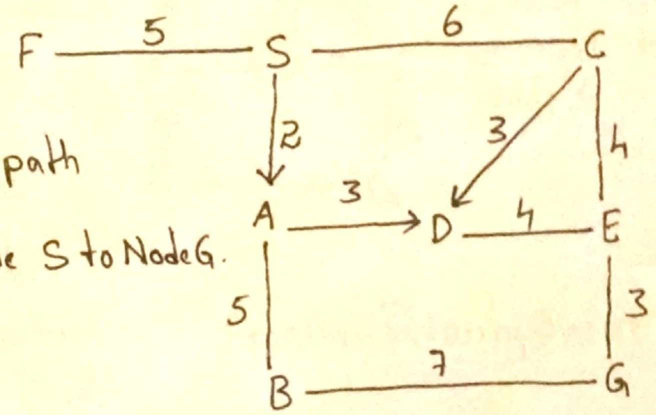
PROBLEM - III   UNIFORM-COST GRAPH SEARCH

Given Graph:



Required: Travel path from Node S to Node G.

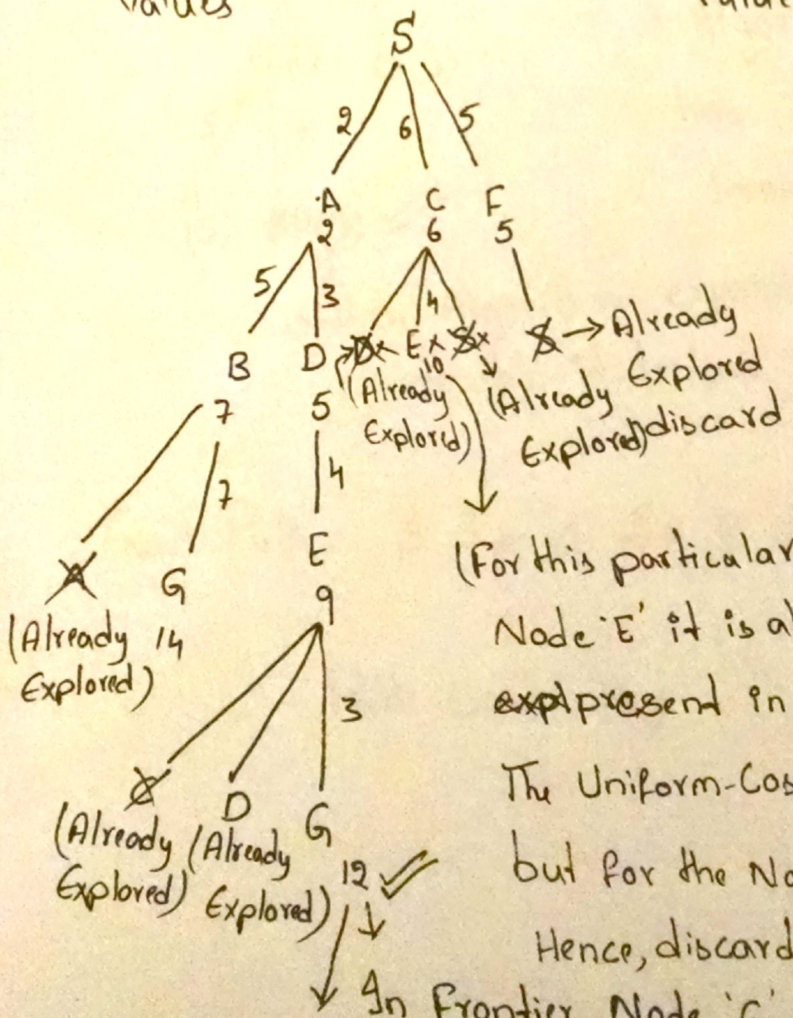
Initial Node: S.
Goal State: G.
Frontier Used: Priority Queue.

We Calculate using g(n) values

Lowest Value to explore

Frontier	~~A~~	~~C~~	~~F~~	~~B~~	~~D~~	~~E~~	G			
Value	2	6	5	7	5	9	~~14~~			

12 → Replace with new value



(For this particular Node 'E' it is already ~~exp~~ present in Frontier The Uniform-Cost Value for this is '10' but for the Node 'E' in frontier it is '9' Hence, discard)

In Frontier Node 'G' has Uniform Cost as 14 hence replace the frontier with this Node value

Goal reached. Hence, Uniform Cost Value = 12//

D&F has same Cost value, hence We go by alphabetical Notion.

Uniform-Cost Value = 12//

Explored Set

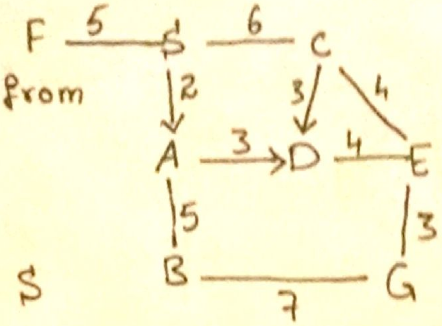
S
A
D
F
C
B
E

Path: $S \xrightarrow{2} A \xrightarrow{3} D \xrightarrow{4} E \xrightarrow{3} G$

Path Cost: 12//

PROBLEM-IV : GREEDY-BEST-FIRST TREE SEARCH

Given:



Required: travel path from Node 'S' to 'G'

Heuristic Values for each Node to travel to Node 'G'

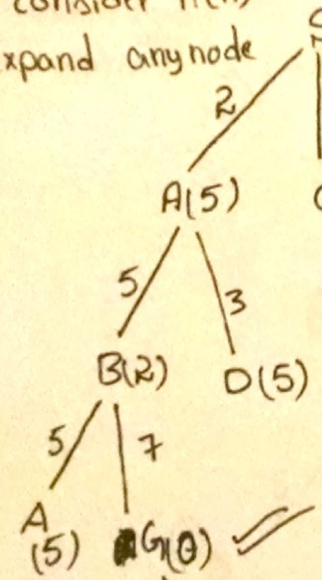
$h(G)$ → Goal Node.

A : 5
B : 2
C : 7
D : 5
E : 2
F : 10
G : 0
S : 8

Initial Node: S

Goal Node: G.

* We consider h(n) values to expand any node



→ When Algorithm expands this particular node then it declare Goal reached.

Frontier: Priority Queue

Node →	~~A~~	C	F	~~B~~	D	~~G~~	A
heuristic Value →	5	7	10	2	5	0	5

* Algorithm expands a node which has low heuristic value in the Frontier.

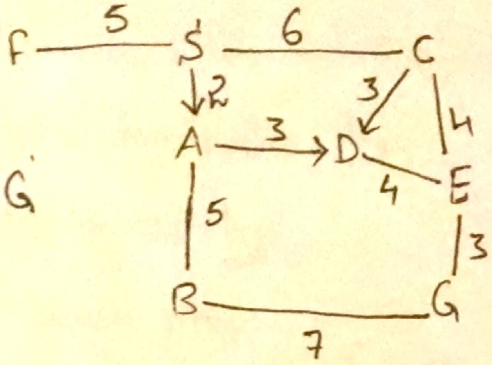
Goal-Path: $S \xrightarrow{2} A \xrightarrow{5} B \xrightarrow{7} G$ (Goal)

Path Cost: 14 //

PROBLEM V

A* TREE SEARCH.

Given Map:



*Required: Path from Node 'S' to Node 'G'

$f(n) = g(n) + h(n)$

↑ path cost ~~from~~ to each node ↑ heuristic value.

Heuristic Values:

h

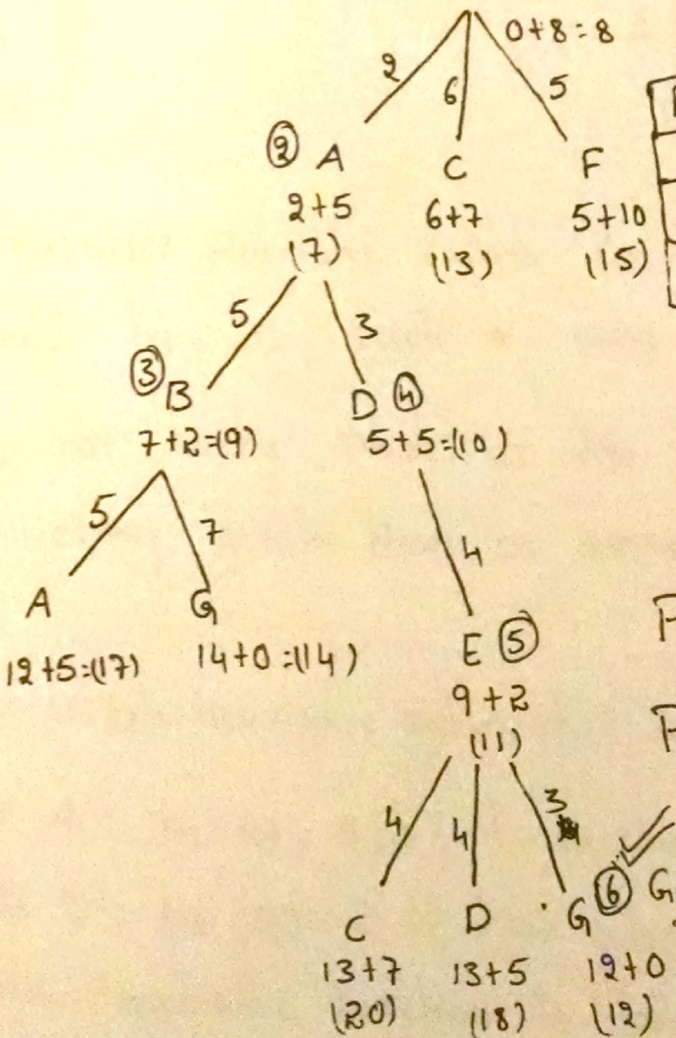
A : 5
B : 2
C : 7
D : 5
E : 2
F : 10
G : 0
S : 8

Frontier Used: **Priority Queue**

Initial Node: S

Goal State: G.

* We expand a node which has least f(n) value in the frontier



$f(n) = g(n) + h(n)$

0 + 8

Frontier	~~A~~ (1)	C	F	~~B~~ (2)	~~D~~ (3)	A	G	~~E~~	C
g(n)	2	6	5	7	5	12	14	9	13
h(n)	5	7	10	2	5	5	0	2	7
f(n)	7	13	15	9	10	17	14	11	20

	D	~~G~~ (4)	
g(n)	13	12	
h(n)	5	0	
f(n)	18	12	

⑥ Goal Reached

Path Cost: 12//

Path: $S \xrightarrow{2} A \xrightarrow{3} D \xrightarrow{4} E \xrightarrow{3} G$

(Goal Reached)

* When algorithm expands Node 'G' it declare Goal Reached. Node 'G' is already added to the frontier it doesn't consider as the f(n) value is '14' which is greater than the optimal path Node G (12)

PROBLEM - VI

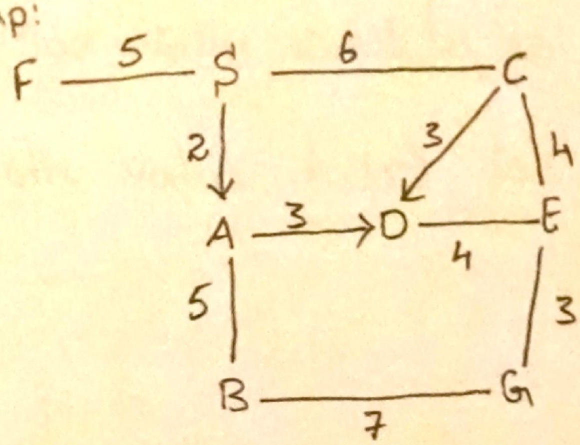
ADMISSIBLE HEURISTIC FUNCTION H^*

Required: To Come up With Admissible Heuristic Function h^* i.e; h_1 i.e., dominates h_2.

Heuristic Function Values h_2

$$\left.\begin{array}{l} A : 5 \\ B : 2 \\ C : 7 \\ D : 5 \\ E : 2 \\ F : 10 \\ G : 0 \\ S : 8 \end{array}\right\} h_2$$

Given Map:



* Admissible Heuristic Values for h_1 which dominates h_2 should be in such a way that $h_1(n) \geq h_2(n)$

i.e; For every node 'n' the heuristic value in h_1 should be either greater than or equal to the heuristic function value h_2.

=> Admissible Heuristic Values (h_1): Assumptive values from Each node to Goal State.

Node 'A': $h_2(A) = 5$ => Let us Consider $h_1(A) = 8$.

Node 'B': $h_2(B) = 2$ => Max Value could be '7' if $h_1(B) > 7$ then the heuristic function is over estimating the value which is not true. Let's consider it as $h_1(B) = 7$

Node 'C': $h_2(C) = 7$ => Let us consider $h_1(C) = 7$. Max Value = '7' otherwise it is over estimating.

Node 'D' : $h_2(D) = 5 \Rightarrow$ Max value could be '7' $h_1(D) = 6$.

Node 'E' : $h_2(E) = 2 \Rightarrow$ Max Value could be '2' Let $h_1(E) = 2$.

Node 'F' : $h_2(F) = 10 \Rightarrow$ Max value could be '15' Let $h_1(F) = 15$

Node 'G' : Goal State $h_1(G) = 0$ & $h_2(G) = 0$

Node 'S' : $h_2(S) = 8 \Rightarrow$ Max Value Could be '12' Let $h_1(S) = 10$

Hence, The new Heuristic value $h_1(n)$ is:

h_1	h_2
A: ~~6~~8	A: 5
B: 7	B: 2
C: 7	C: 7
D: 6	D: 5
E: 2	E: 2
F: 12	F: 10
G: 0	G: 0
S: 10	S: 8

Sum of all the values $h_1(n) = 52 >$ Sum of all the values $h_2(n) = 39$.

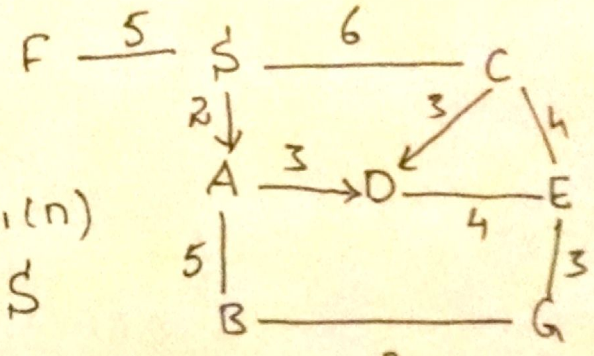
Hence h_1 dominates h_2.

When an A^* Search is Implemented using h_2 the no. of nodes expanded are = 6 nodes

Let us Implement A^* Search using the new Heuristic function values.

③ A* Tree Search Using new Heuristic Function Values.

Given Map:



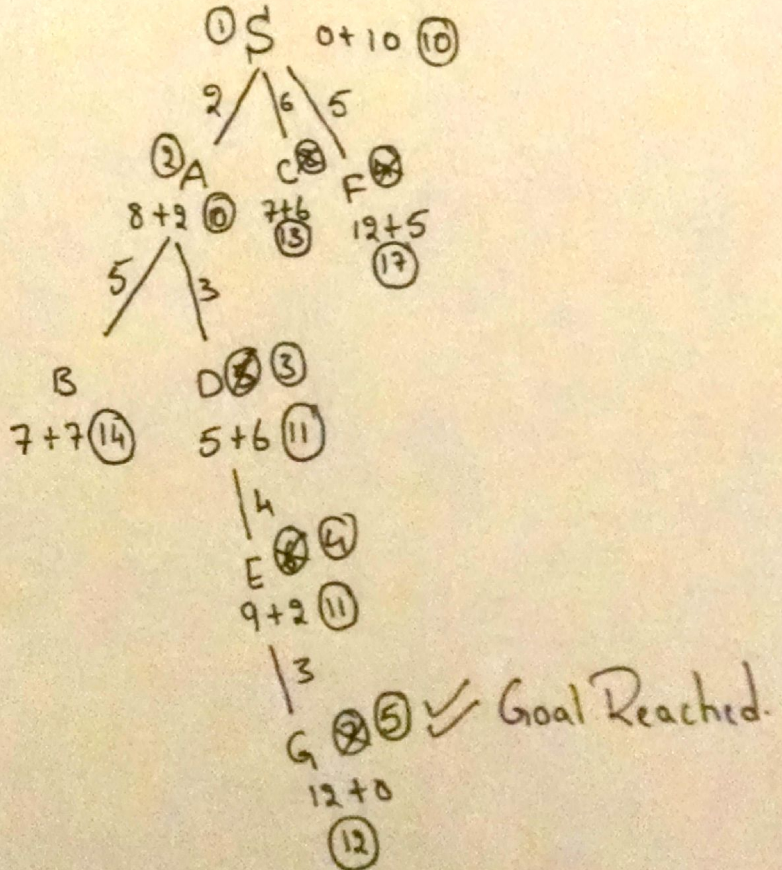
$f(n) = g(n) + h_1(n)$

Initial Node: S

Heuristic Values $h_1(n)$

A : 8
B : 7
C : 7
D : 6
E : 2
F : 12
G : 0
S : 10

Frontier	~~A~~	C	F	B	~~D~~	~~E~~	~~G~~				
g(n)	2	6	5	7	5	9	12				
h(n)	8	7	12	7	6	2	0				
f(n)	10	13	17	14	11	11	12				



Total No of Nodes Expanded : 5// Hence $h_1(n) \geq h_2(n)$

h_1 dominates h_2//