Assignment NO 1 part- A

Name:- Mahesh Maruti Malusare

Class:- B.E.

Roll NO:- 33

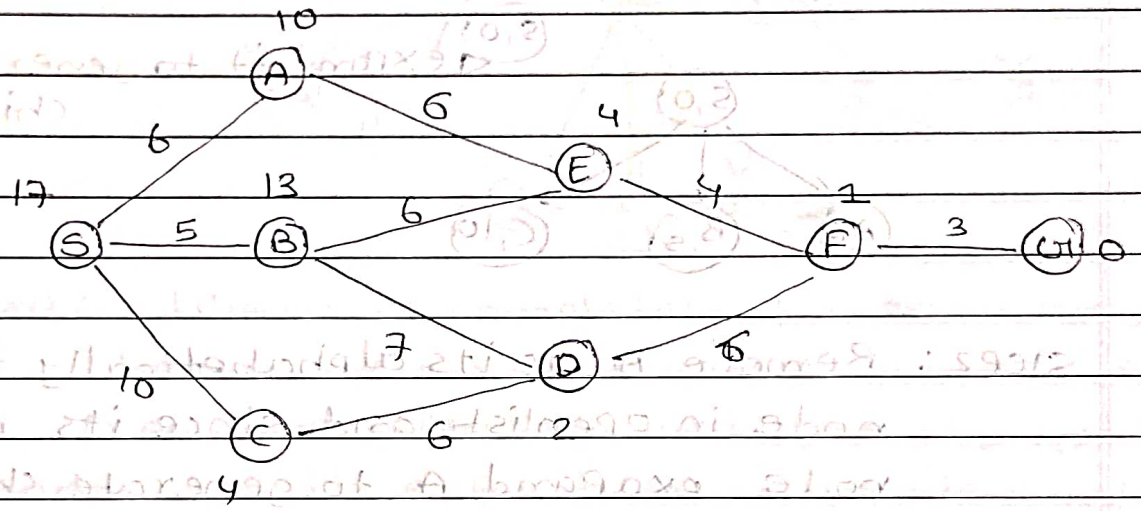
Branch:- I.T.

Sub:- AI

DOP	DOC	marks	sign

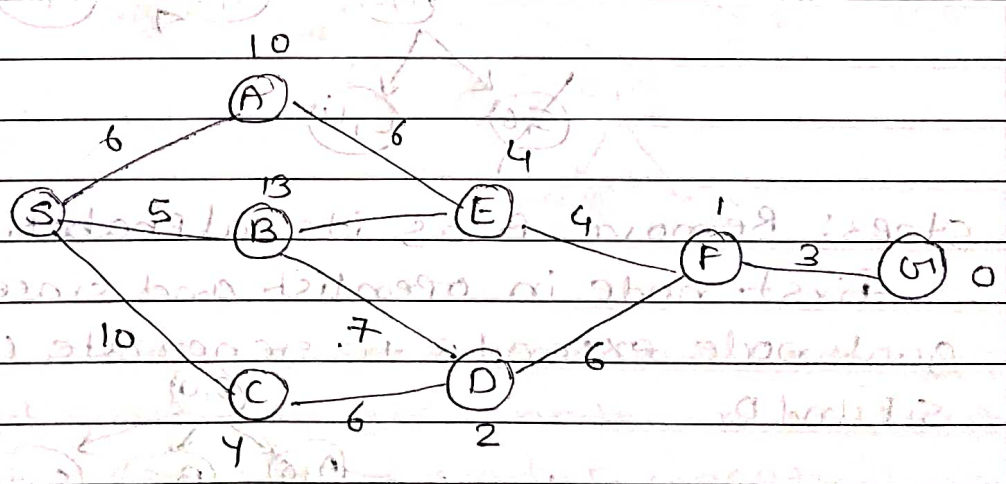
PAGE No.
DATE / /

Q1 Consider following definition of state space for some arbitrary problem. The number mentioned against the edges is cost to be incurred in moving from one node to the other in any direction. The number in red font mentioned against the node is the heuristic function value.



using above data solve following problem:

Q1.1 Apply BFS on above graph

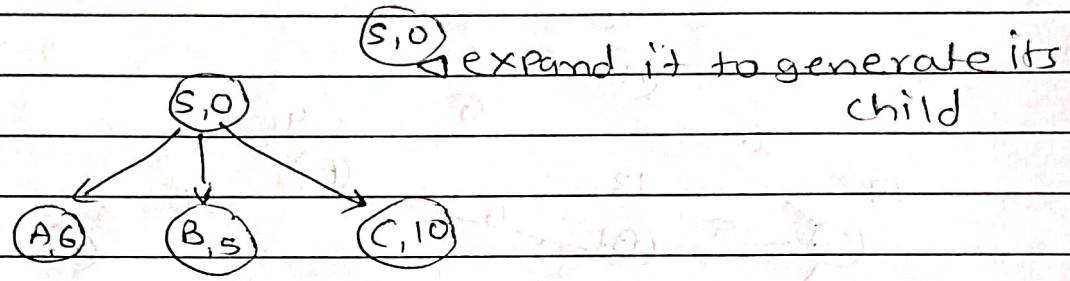


BFS (Breadth first search is a uninformed search algorithm) we will show step in

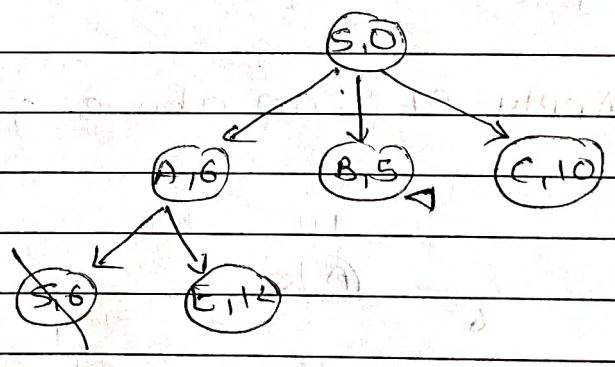
Graph search implementation of this algorithm on above problem:

Step 0: Put initial node S into openlist path cost for it is 0

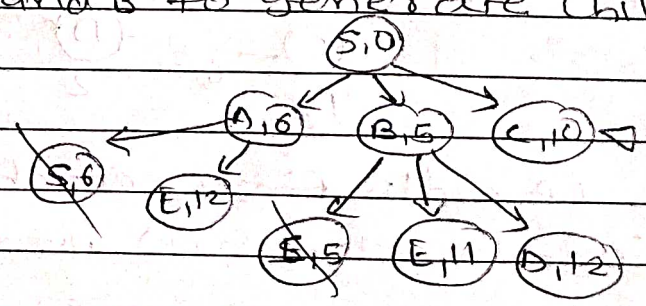
Step 1: Remove S from openlist and since its not goal node nodes A, B, C.



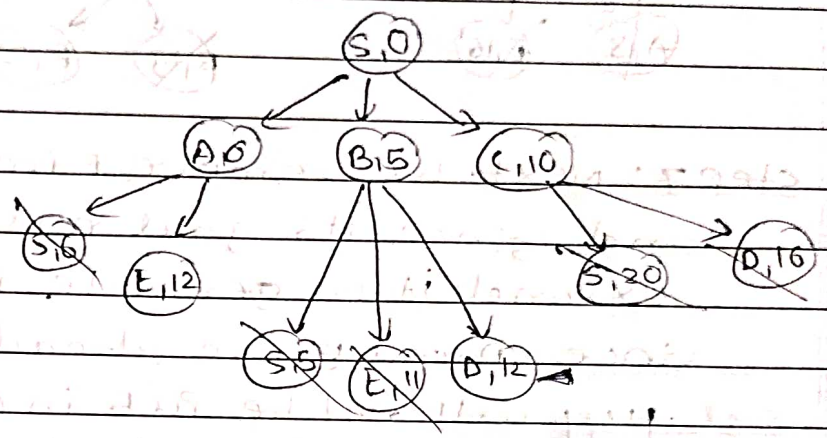
Step 2: Remove A as its alphabetically first node in openlist and since its not goal node expand A to generate child nodes S and E.



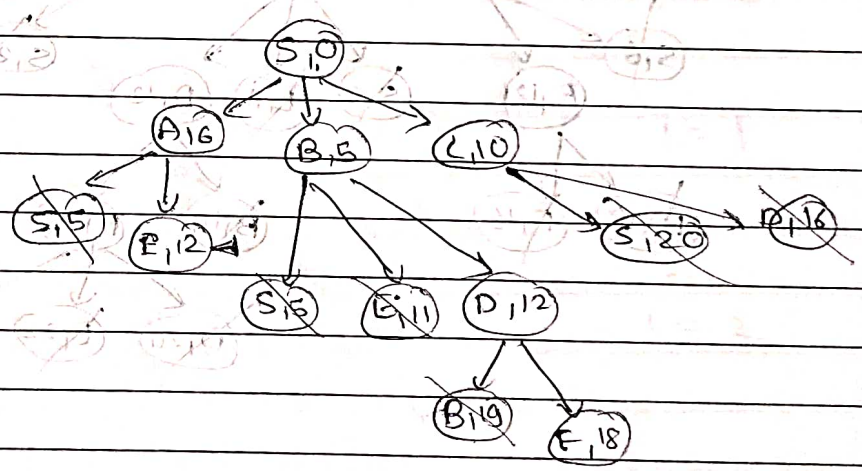
Step 3: Remove B as its alphabetically first node in openlist and since its not goal node expand B to generate child nodes S, E and D.



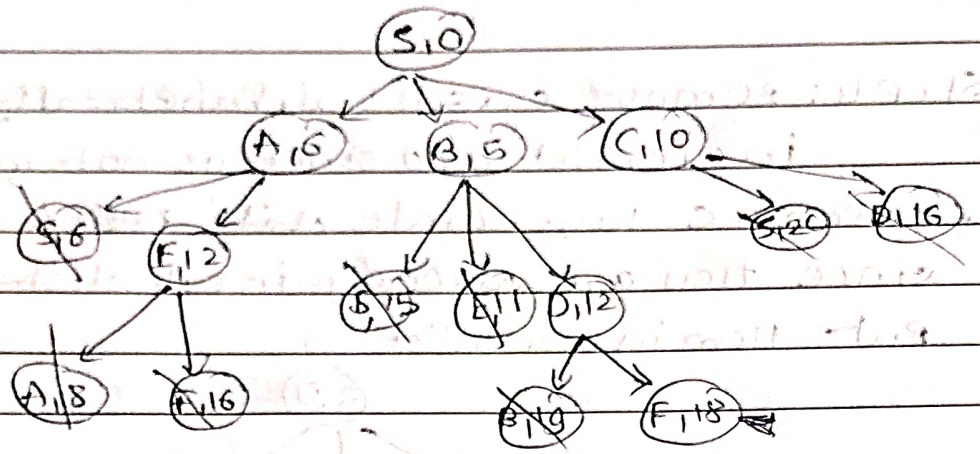
Step 4: remove C as its alphabetically first node in openlist and since its not goal node expand C to generate child nodes S and D since they are already in openlist dont put them in openlist.



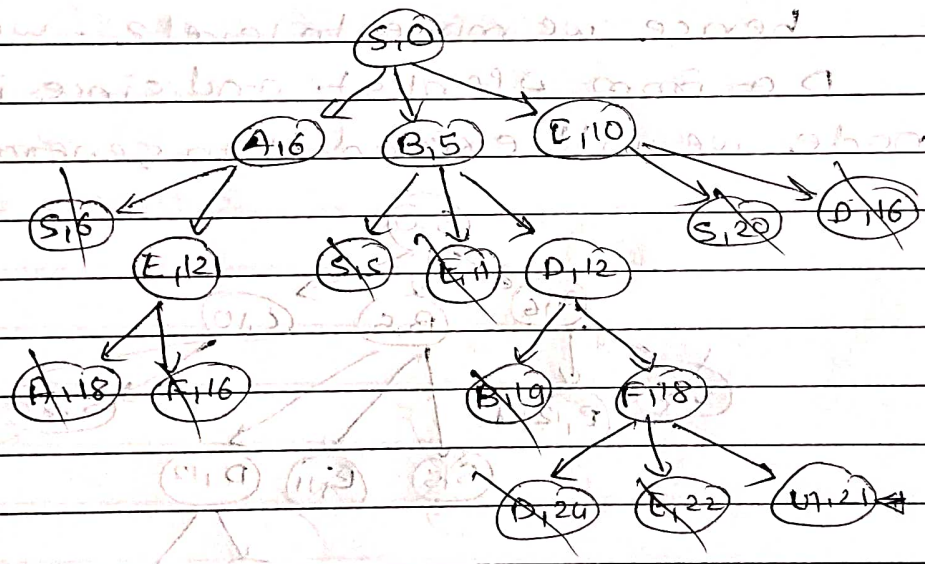
Step 5: level 1 is completely traversed and hence we move to level 2. we remove D from openlist and since its goal node we will expand it to generate B and S



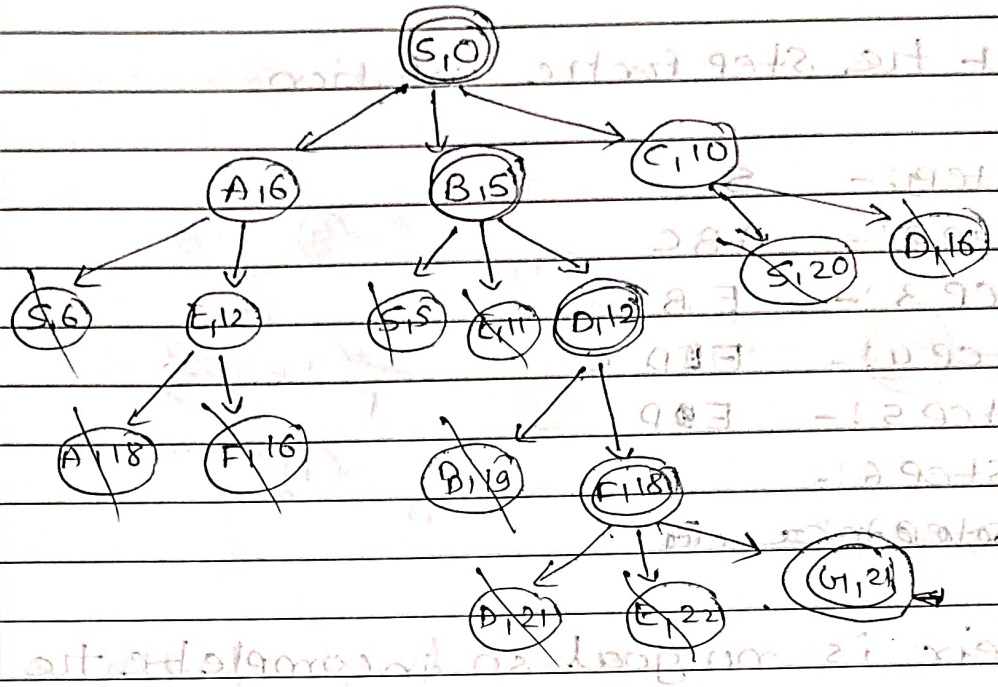
Step 6: Next we remove E from openlist and since its goal node we will expand it to generate A and F since they are already expanded they will not be put in openlist.



Step 7: Next we remove F from openlist and, since its goal node we will expand it to generate, D E and G since D and E are already expanded they will not be put in openlist only G is send to

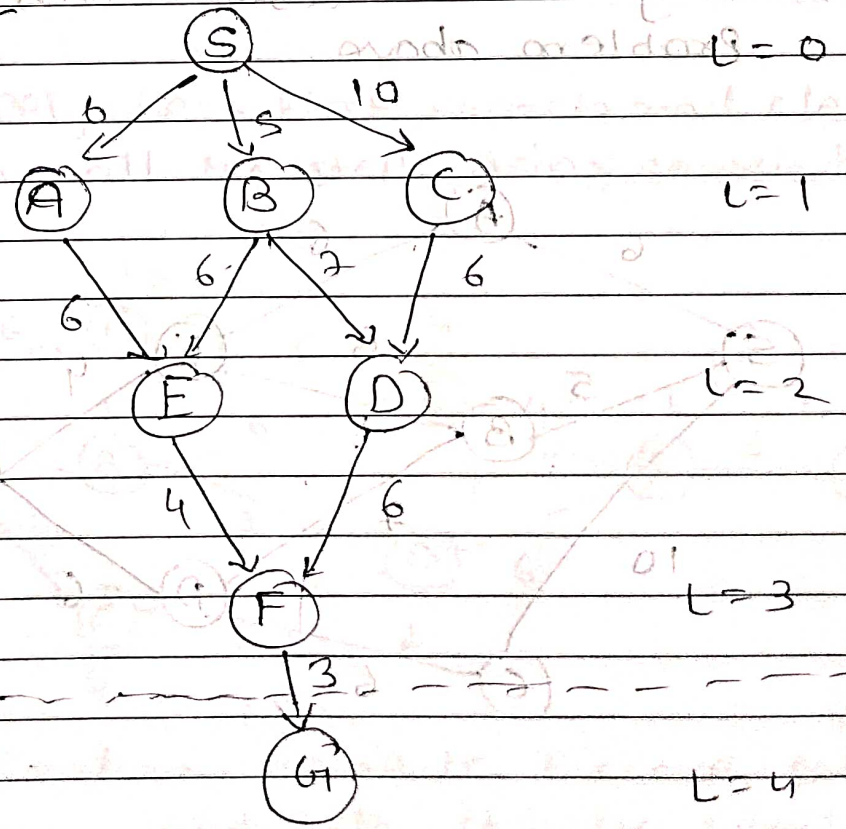


Step 8: Next we remove G from Openlist G is goal node and hence we stop search and return solution by transversing back to root node and reversing the path starting at G.



Q1.2 Apply Depth limited DFS with $l = 3$ for this algorithm to be complete what should be the minimum value of l.

solution:-

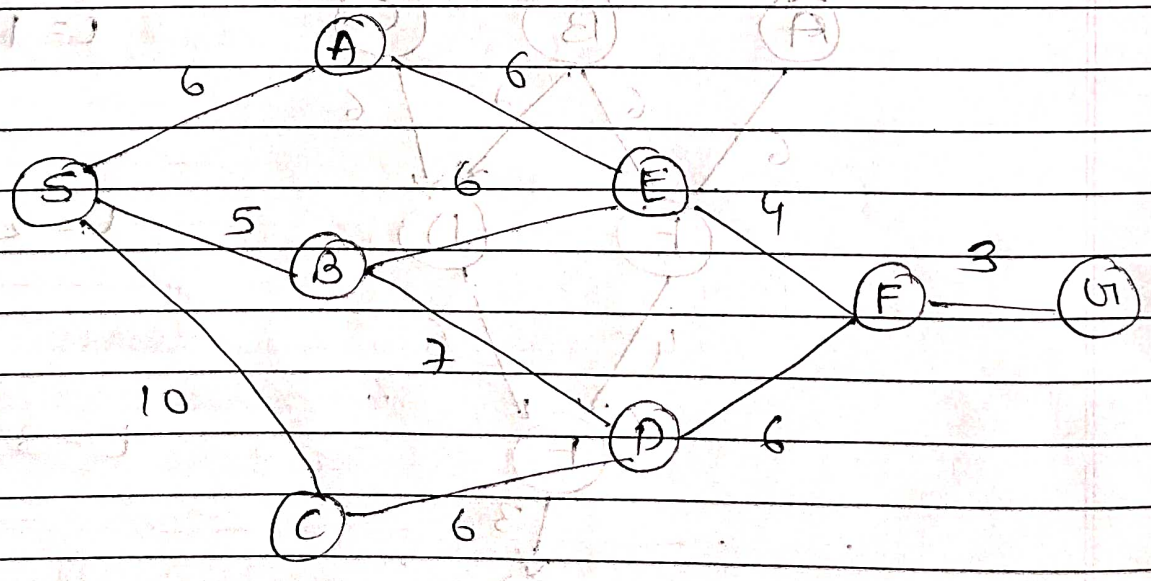


let the step for the solution

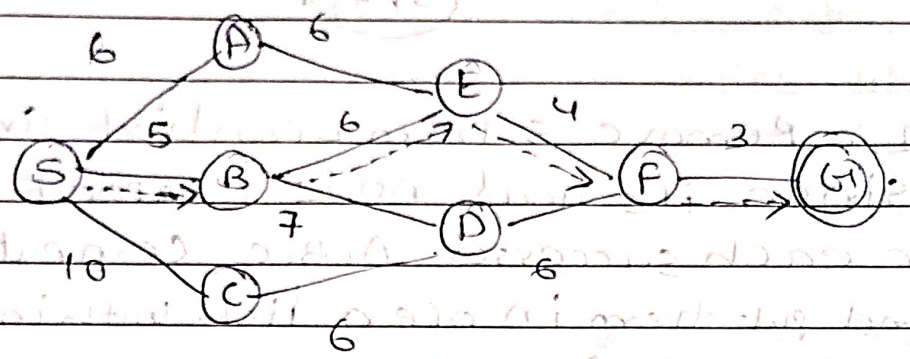
Step 1:- S.
step 2:- A B C
step 3:- E B C
step 4:- E D C
step 5:- E D
step 6:- F

Their is no goal so incomplete the solution at the $l=3$
for the complete we have to $l=4$ for the achieve goal node.

Q1.3 Apply uniform cost search on the Problem above.



Solution:

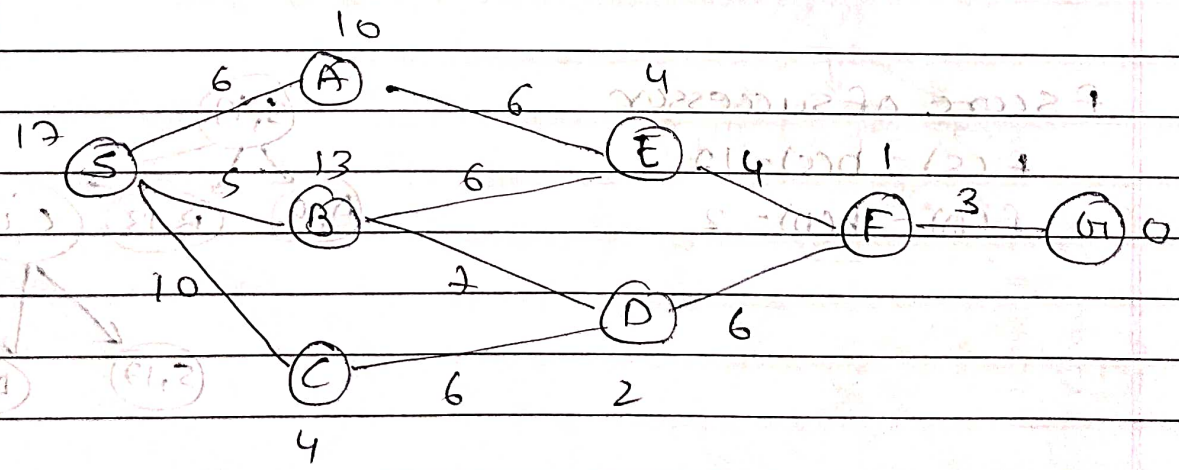


Path to goal node

$S \xrightarrow{5} B \xrightarrow{6} E \xrightarrow{4} F \xrightarrow{3} G$

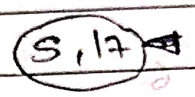
solution cost is 18.

Q1.4 Apply Best First search and clearly show all the steps using search tree



Initialization: compute f-score for s and put it in the openlist.

F score S: $F(S) = h(S) = 17$



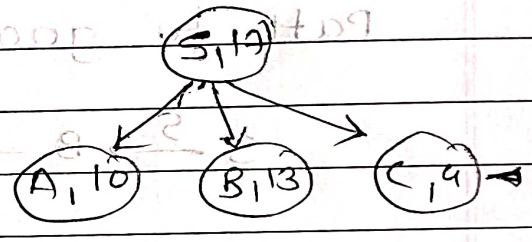
Step1: Remove S from openlist and since its goal node expand it for each successor A, B, C compute f score and put them in open list with increasing order of f-score

f score of successor

$f(A) = h(A) = 10$

$f(B) = h(B) = 13$

$f(C) = h(C) = 4$

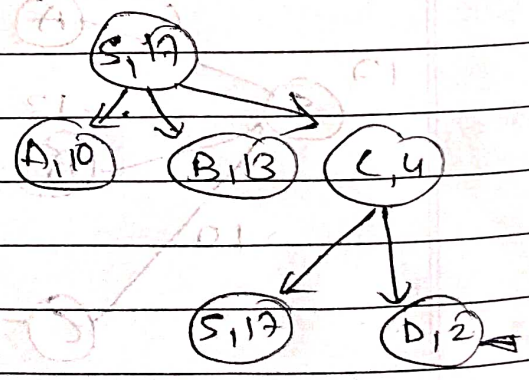


Step2: Remove C from openlist and since its goal node expand it for each successor S, D, compute f score and put them in open list with increasing order f-score

f score of successor

$f(S) = h(S) = 17$

$f(D) = h(D) = 2$



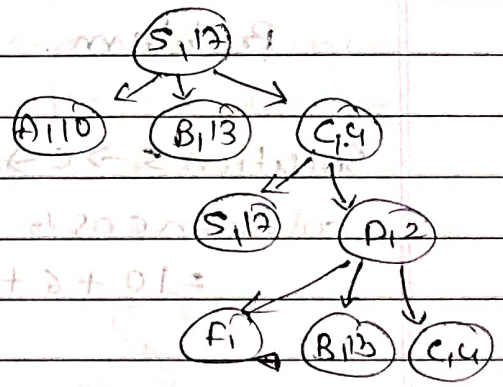
Step 3: Remove D from openlist and since its goal node expand it. For each successor C, B, F compute f score and put them in openlist with increasing order of f-score

F score of successor

$F(C) = h(C) = 4$

$F(B) = h(B) = 13$

$F(F) = h(F) = 1$



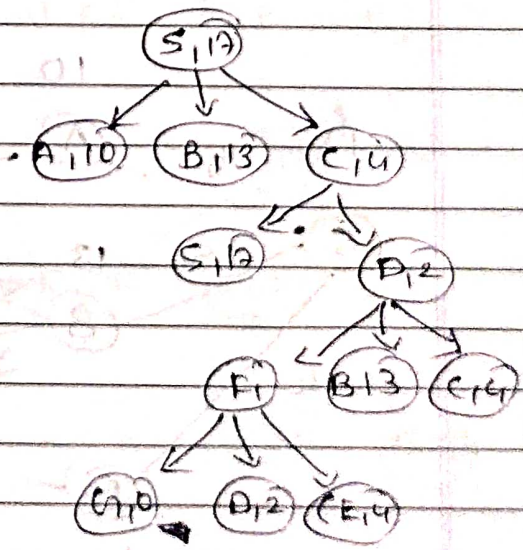
Step 4: Remove F from openlist and since not goal node expand it for each successor D, E, G compute f score and put them in open list with increasing order of f-score

F score of successor

$F(D) = h(D) = 2$

$F(E) = h(E) = 4$

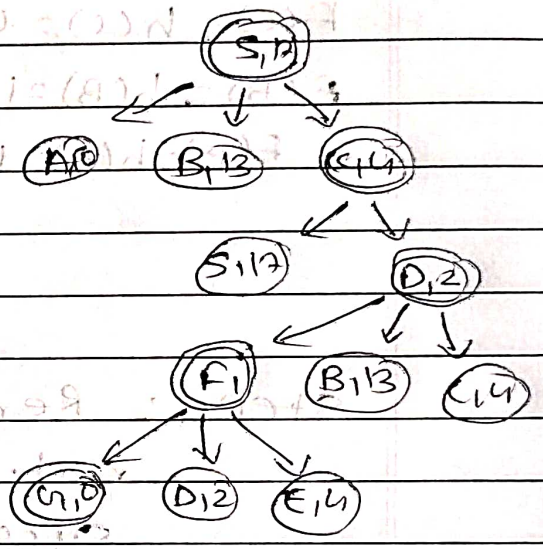
$F(G) = h(G) = 0$



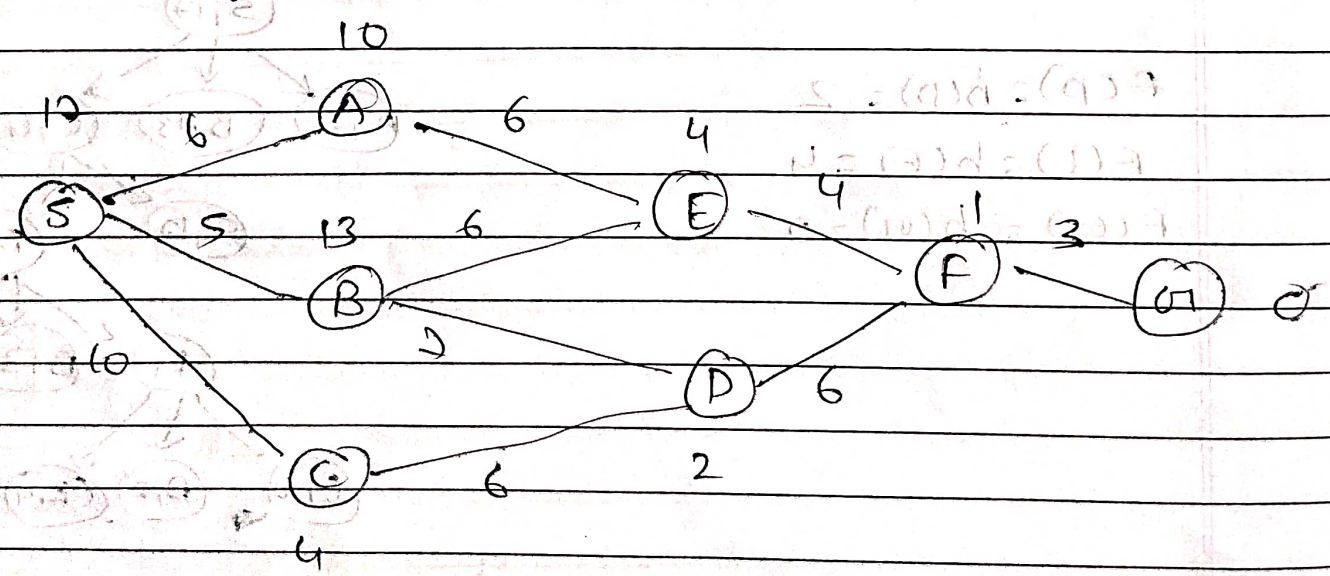
Step 5 Remove G from openlist and since its goal node it passes goal test stop search and traverse backwards following parent pointers to the root node of search tree reverse the path traversed return it as solution to problem.

Solution S → C → D → F → G with solution cost

$= 10 + 6 + 6 + 3 = 25$



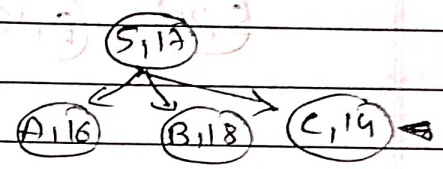
Q1.5 Apply A* algorithm and clearly show all steps using search tree



Step1:- calculate f score for S (F(S)=0+13=17)
Put S in open list

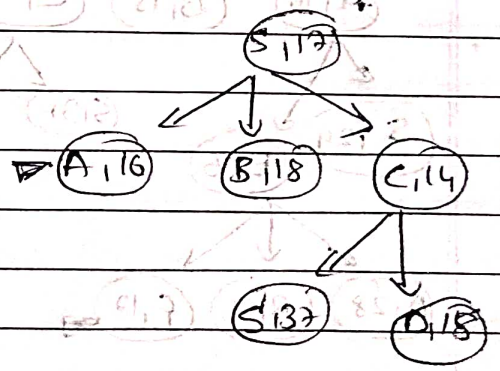
Step2:- get S out of openlist and since its goal apply successor function to get A, B and C as its successors
Compute f score for them and put them in openlist in Priority order

F(A) = 6+10 = 16
F(B) = 5+13 = 18
F(C) = 10+4 = 14

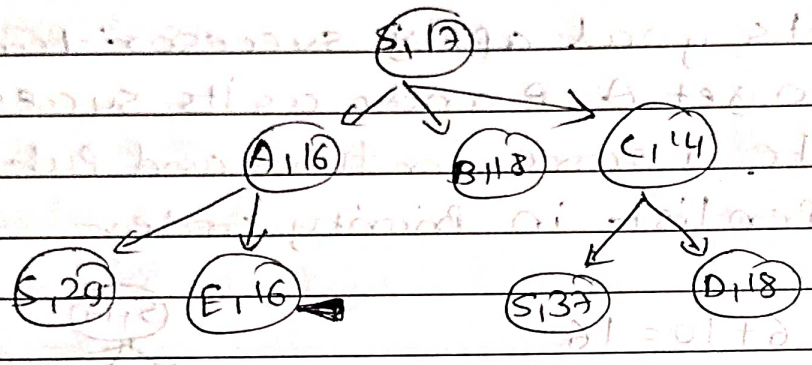


Step3:- get C out of open list and since its not goal apply succesor function to get S, D as its successor
compute F score for them and put them in openlist in Priority order

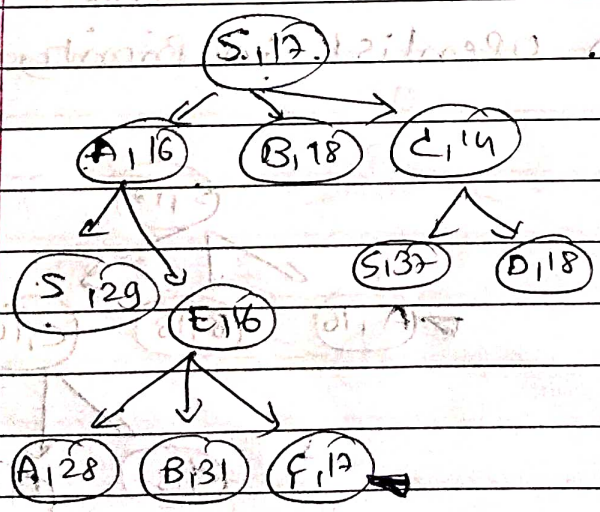
F(S) = (6+6)+17 = 129
F(D) = (6+6)+4 = 16



Step4:- get A out of open list and since its goal apply successor function to get S, E as its successor compute f-score for them and put them in Openlist in Priority order.



Step5: get E out of open list and since its not goal apply sucessor function to get A, B, and F as its successor compute f score for them and put them in Openlist in Priority order

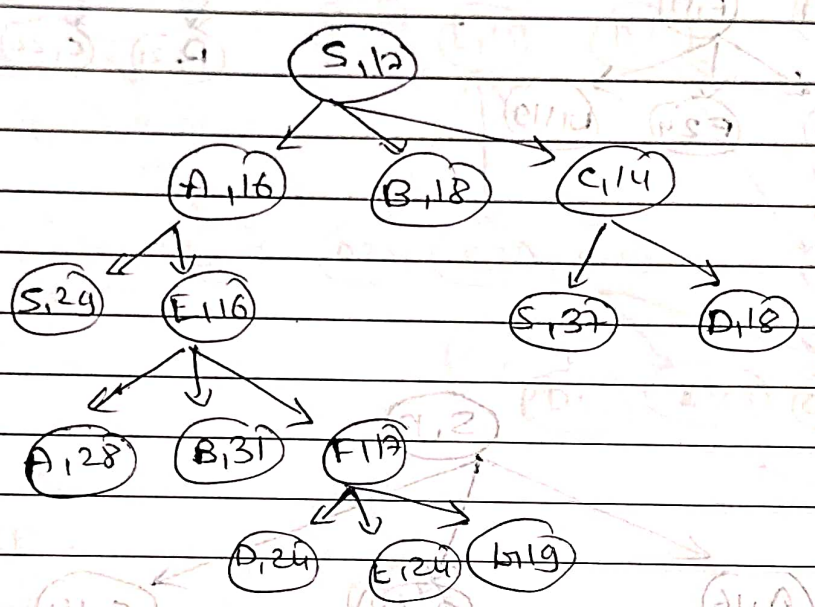


$F(A) = (6+6+6) + 10 = 28$

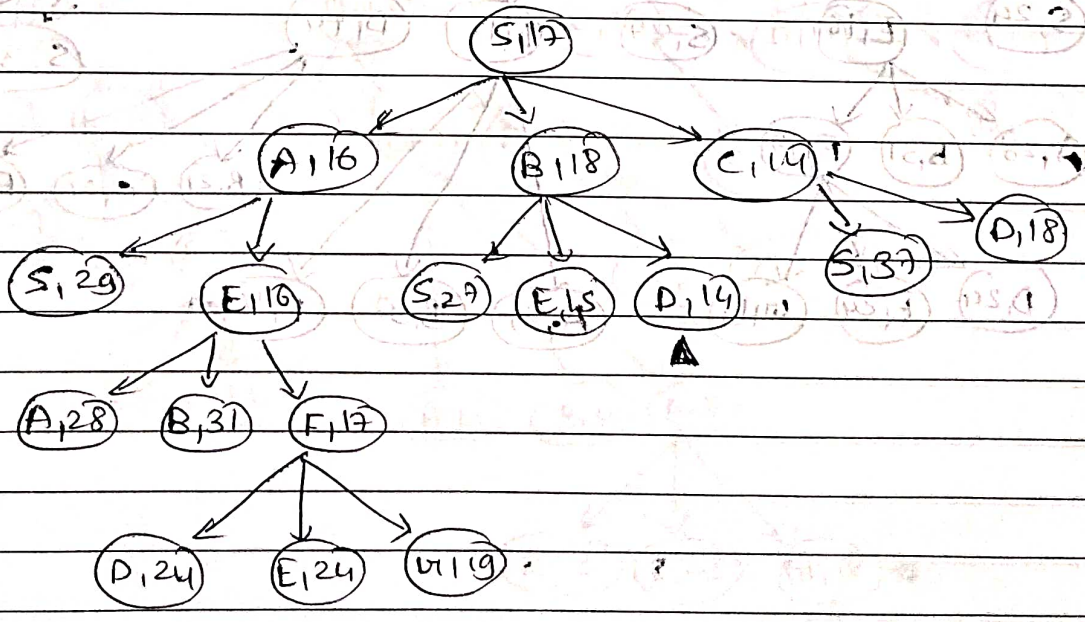
$F(B) = (6+6+6) + 13 = 31$

$F(F) = (6+6+4) + 1 = 17$

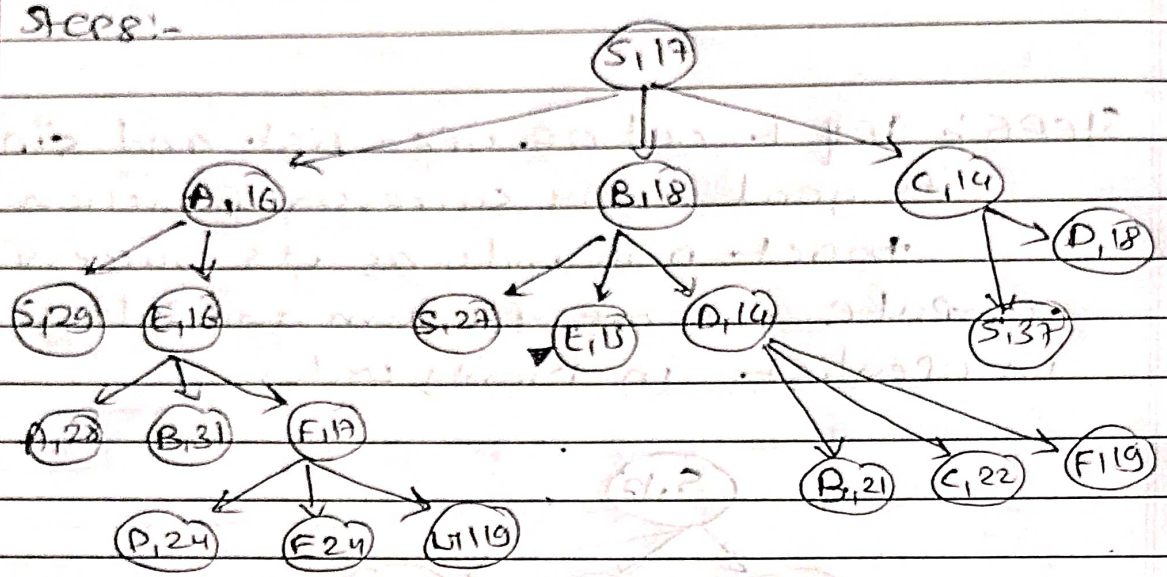
Step6 :- get F out of open list and since its goal apply successor function to get D, E and G as its successor compute F score for them and put them in openlist in Priority order.



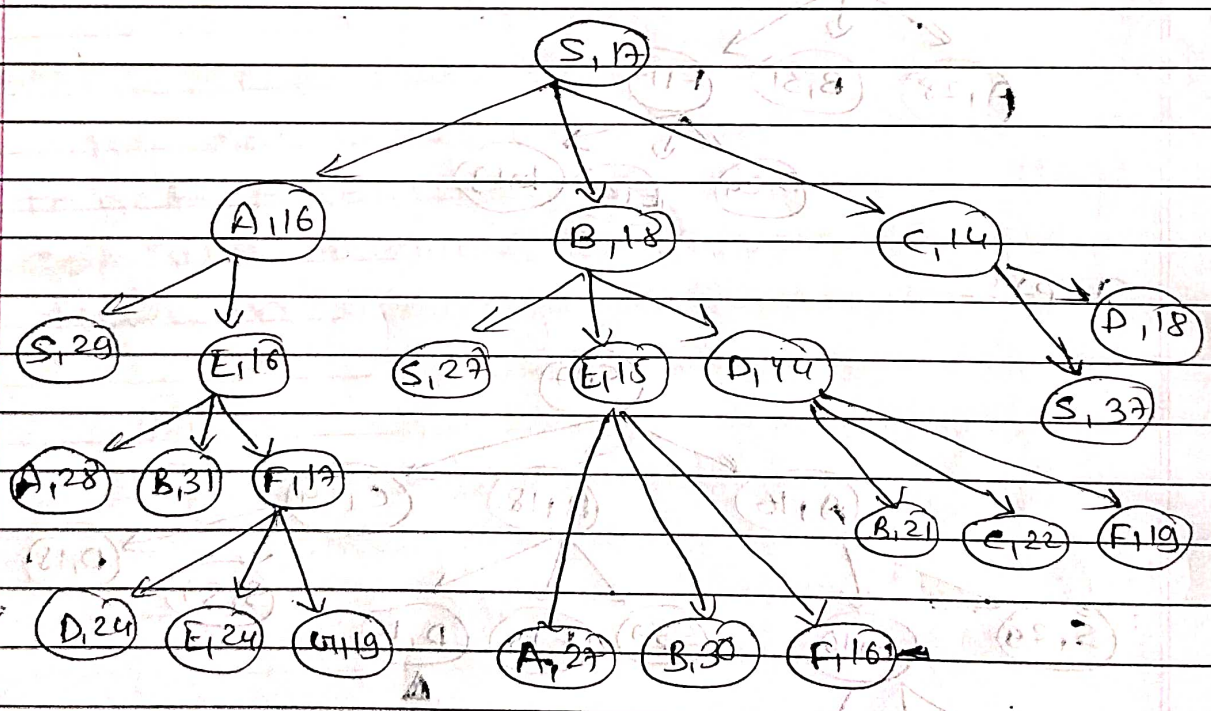
Step7 :-



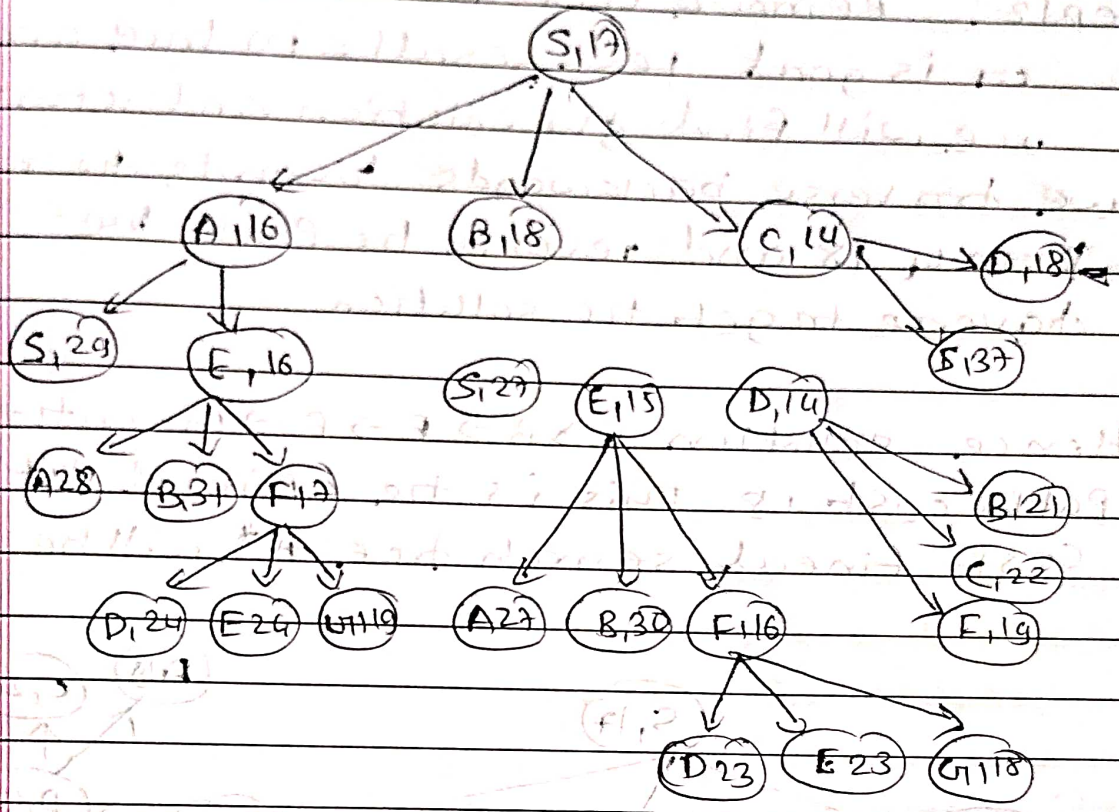
Step8:-


S,17
A,16
B,18
C,14
D,18
S,29
E,16
S,27
E,15
D,14
S,37
A,28
B,31
F,17
B,21
C,22
F,19
D,24
E,24
G,19

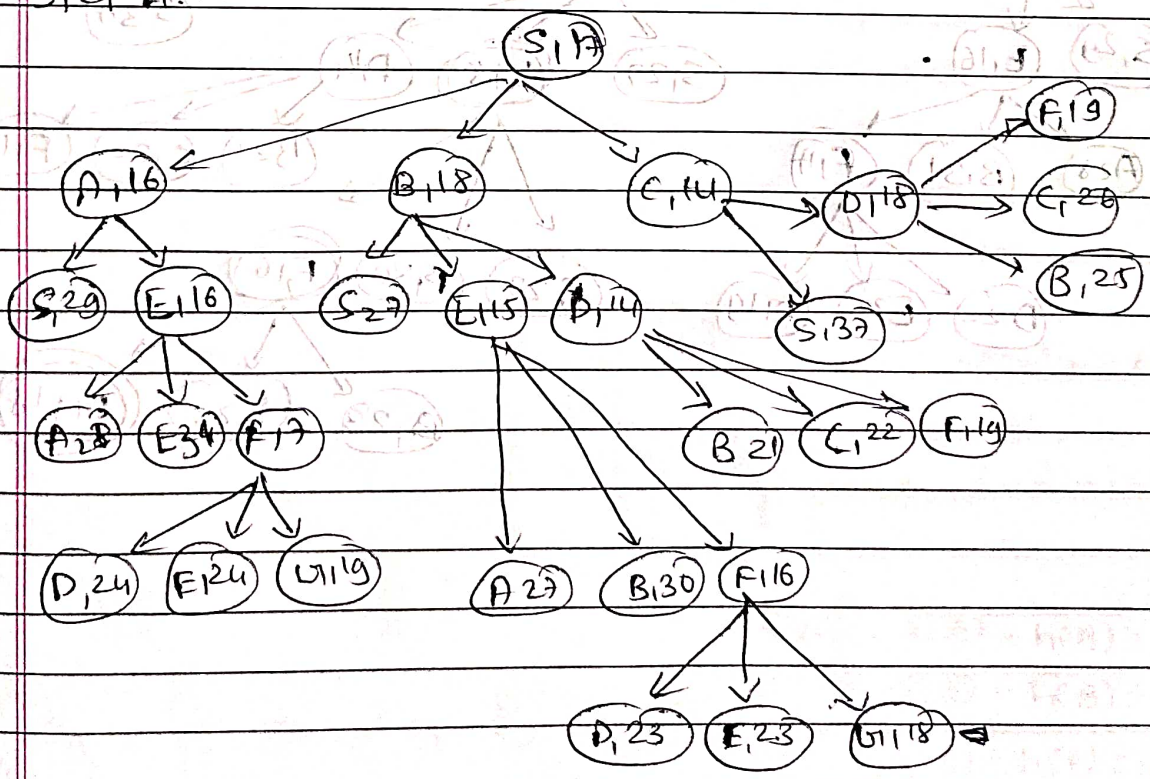
Step9:


S,17
A,16
B,18
C,14
D,18
S,29
E,16
S,27
E,15
D,14
S,37
A,28
B,31
F,17
B,21
C,22
F,19
D,24
E,24
G,19
A,27
B,30
F,16

Step 10 :-

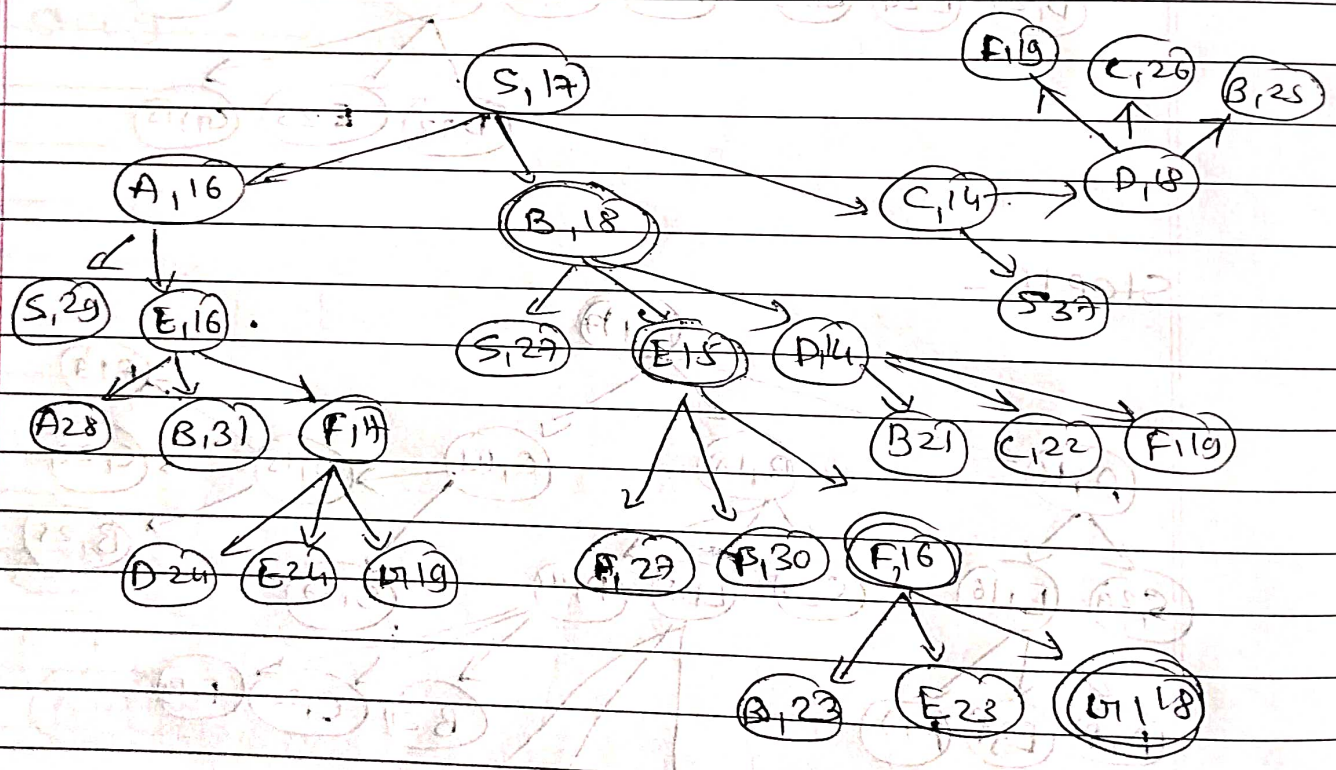


Step 11 :-

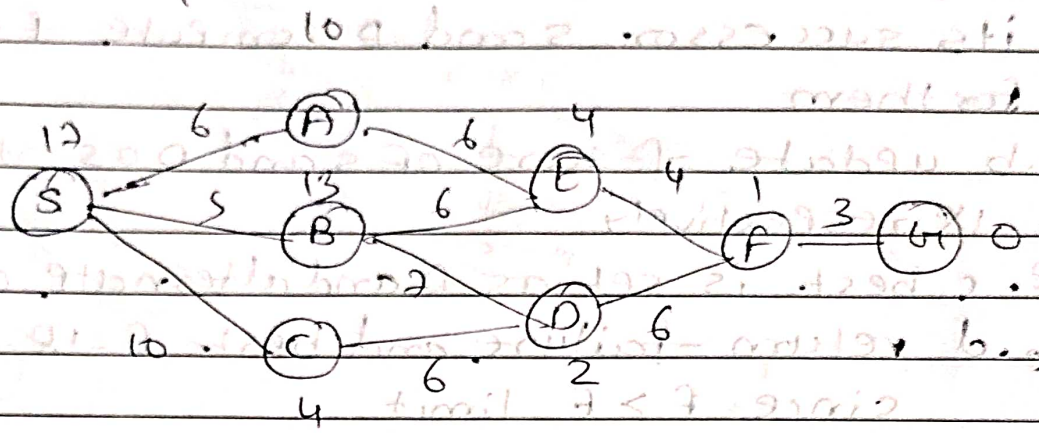


Step 12: Remove G,18 from open list since G is goal test results in true and we will find the solution and return it we traverse backwards towards the root from G 18 and reverse the path we traverse to get the solution

Hence solution S → B → E → F → G with path cost 18 This is the optimal path cost final search tree A* will be



Extra question same problem with RBFS



Step 0: call RBFS with S as initial node and f limit as infinity.

Step 1: RBFS (Problem, S, infinity)

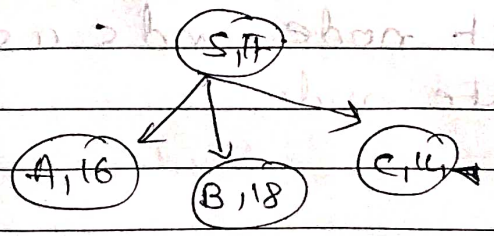
a) since S is not goal node generate its successor A, B and C compute f score for them.

b) If new f-score is > then earlier score (those visited earlier will have f-score computed) or f score initialized first time then f-score for that successor is updated.

c) Best is set as C and alternative as A.

d) update f limit to C.f score = 14

e) call RBFS with problem definition C and min(14, 16)



$F(A) = g(A) + h(A) = 16$

$F(B) = g(B) + h(B) = 18$

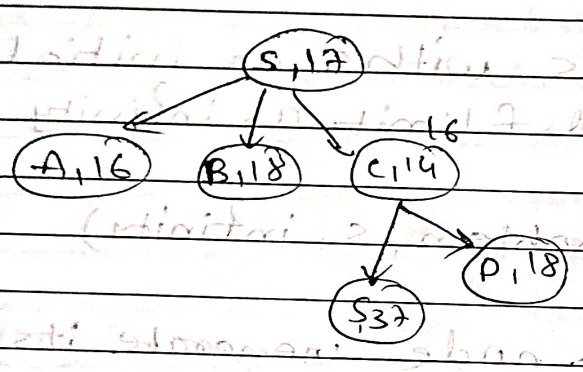
$F(C) = g(C) + h(C) = 14$

e.a since c is not goal node. Generate its successor s and D compute f score for them

e.b update f-score of s and D as 37 and 18 respectively

e.c best is set as D and alternate as s

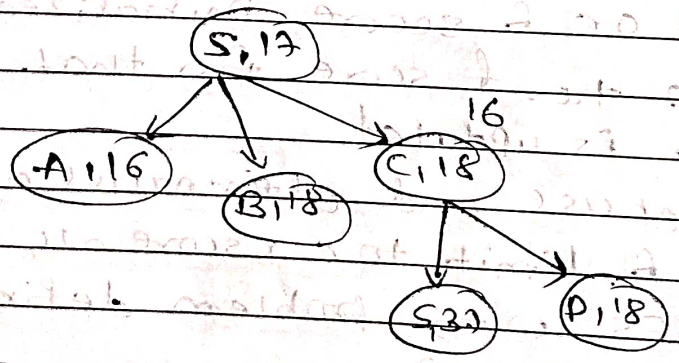
e.d return failure and best f = 18 since f > f limit.



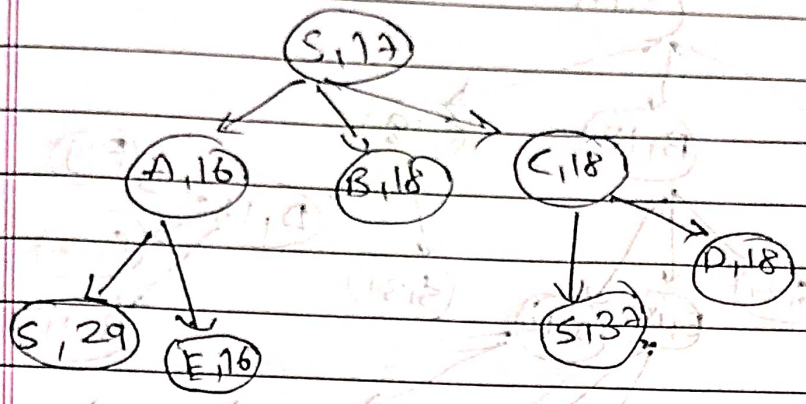
f(S) = (10+10)+17 = 37
since 37 > 17 update S-f score

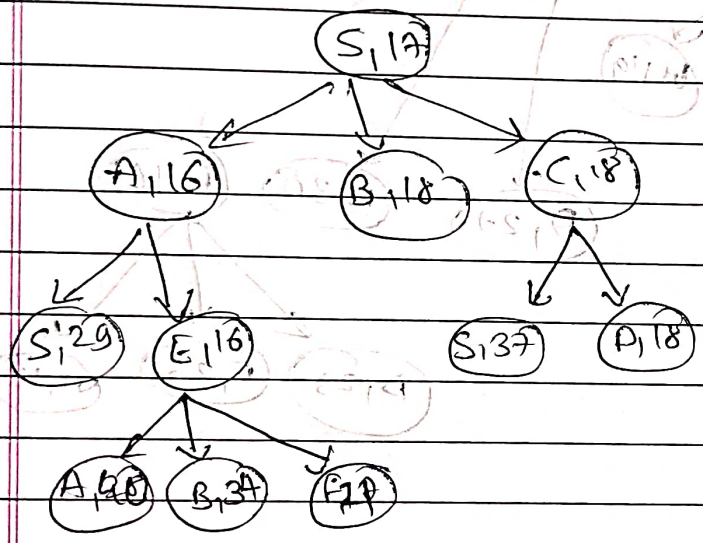
F(D) = (10+6)+2 = 18

f update c fscore as 18 returned from call RBFS with problem definition c and min (14,16)



g. Now A is best node and c as alternate node

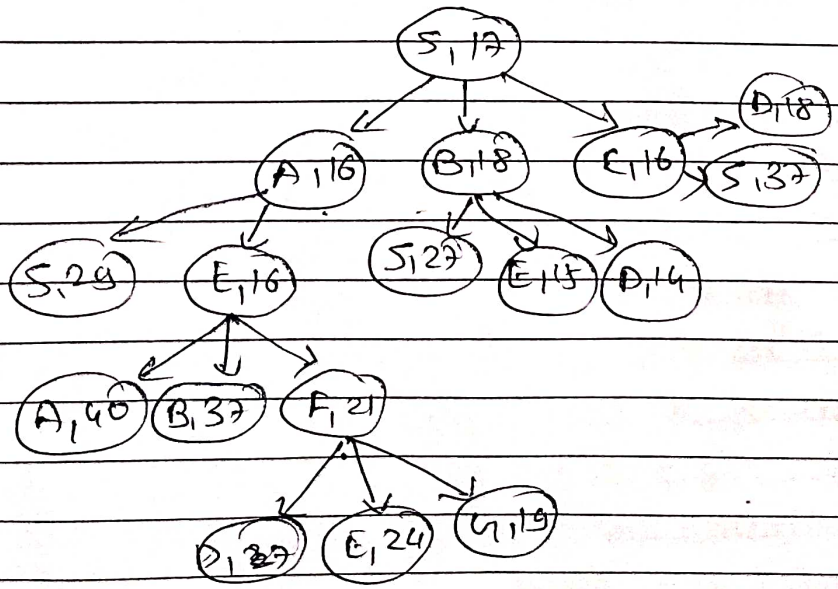


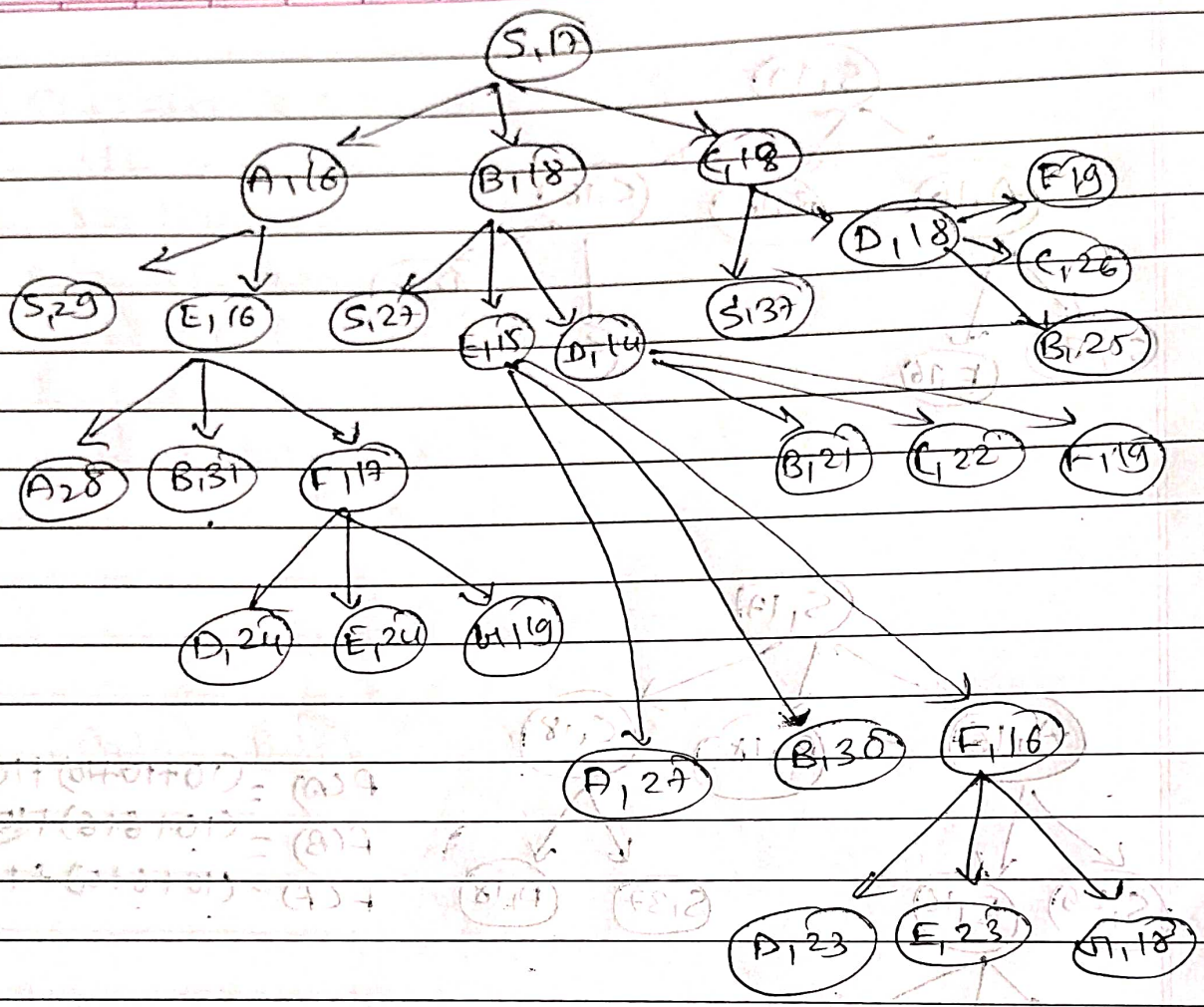


$F(A) = (10+10+10)+10 = 40$

$F(B) = (10+6+6)+10 = 34$

$F(R) = (10+6+4)+1 = 21$




S,17
A,16
B,18
C,18
F,19
D,18
C,26
B,25
S,29
E,16
S,27
E,15
D,14
S,37
A,28
B,31
F,17
B,21
C,22
F,19
D,24
E,24
G,19
D,27
B,30
F,16
D,23
E,23
G,18

PAGE No.
DATE / /

Q2 Consider following instance of 8 puzzle Problem:

8	7	6
2	1	5
3	4	-

Initial configuration

-	8	7
2	1	6
3	4	5

Goal Configuration

consider Heuristic functions defined below:

h_1 : misplaced tiles count except space
h_2 : correctly placed tiles count except space
h_3 : sum of manhattan distance between current and correct position of all times except space.

Answer following question:

a) In 8 puzzle problem we are concerned with getting to goal configuration within least number of steps. All moves are thus costly. Define g(n) in you own words. What will be the cost of 6 step solution to some arbitrary 8 puzzle instance?

→ Misplaced tiles count except space = 3

DATE / /

8	7	6
2	1	5
3	4	-

up

h=1

8	7	6
2	1	
3	4	5

left

h=2

8	7	6
2	1	5
3		4

up

h=1

8	7	
2	1	6
3	4	5

left

h=3

8	7	6
2		1
3	4	5

left

h=1

	8	7
2	1	6
3	4	5

left

	8	7
2	1	6
3	4	5

goal state.

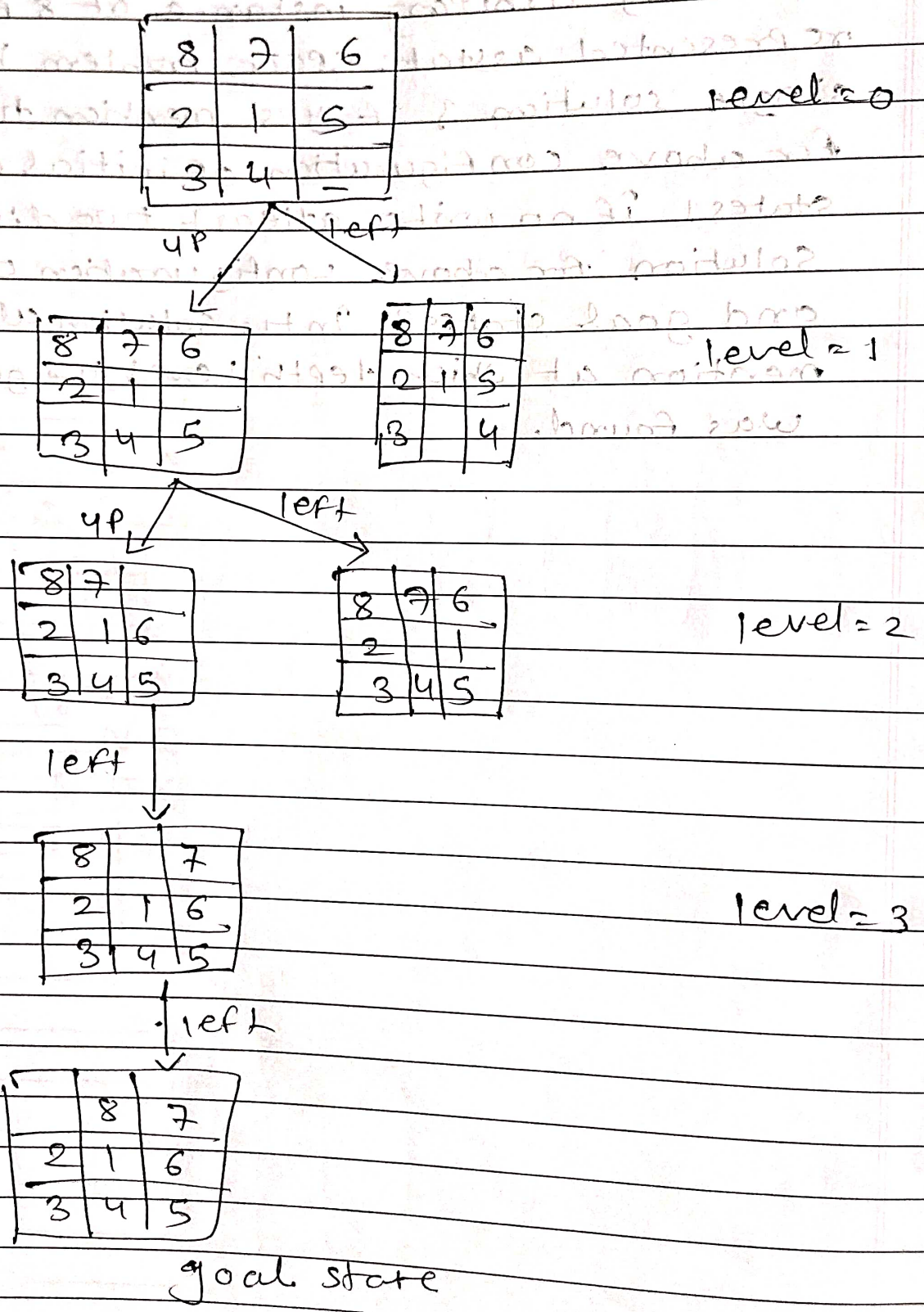
g(n) is the actual cost of the node from the beginning of the path.

b) Does any arbitrary instance of 8 puzzle represented as state space problem has single solution? If yes mention the solution for above configuration of initial and goal states? If no write atleast two distinct solution for above configuration of initial and goal states? In the solution also mention at which depth level the goal was found.

→

c) Draw exhaustive state space tree of depth limited to 4 for instance of 8 puzzle problem in the question.

->



PAGE No.
DATE / /

d) In the tree above name the nodes with letters A, B, C ... starting at leaf level from left to right in horizontal scan line up to initial node. Now for following search strategies mention sequence of visited nodes untill the tree is fully visited

i) Breadth first search

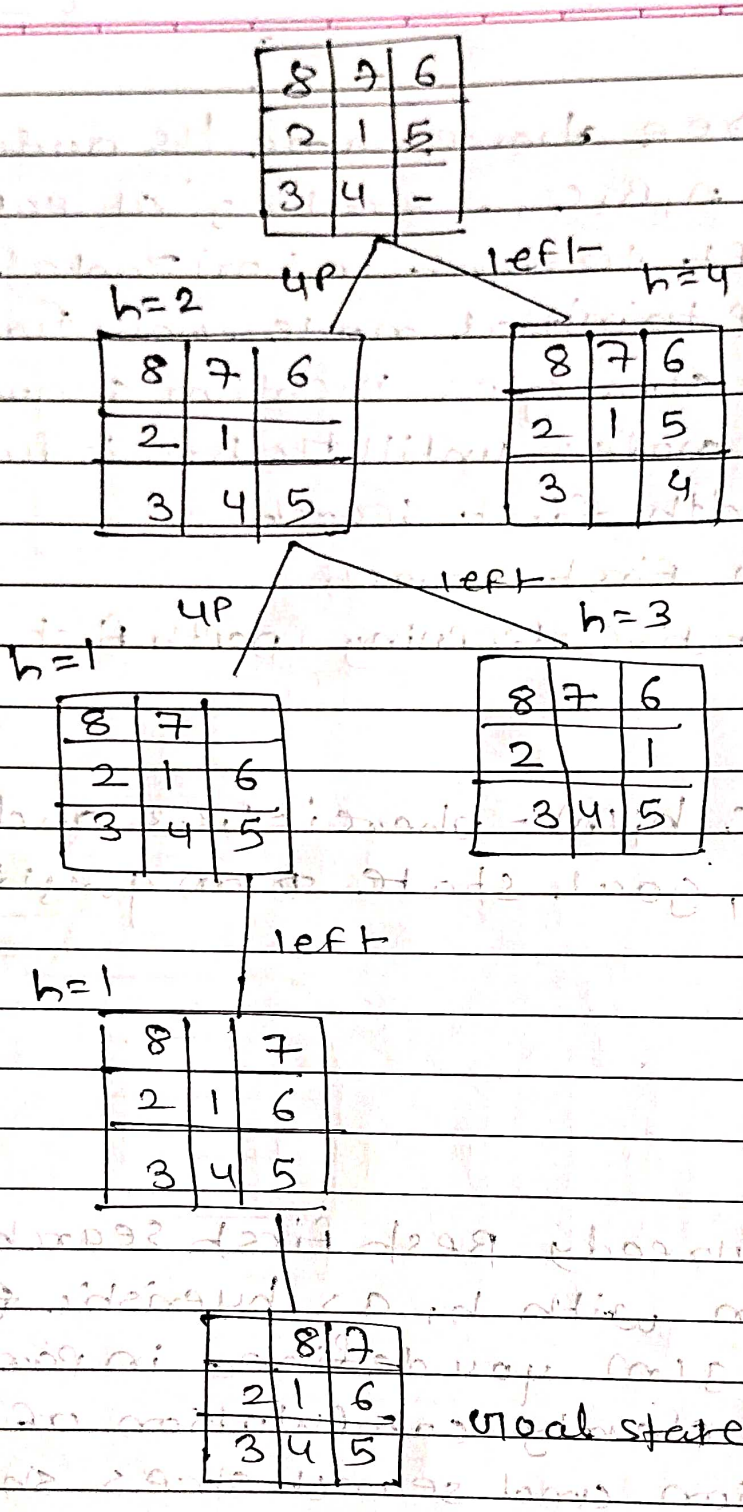
ii) Depth first search

iii) Iterative deepning depth first search

e) compute $h_i(n)$ - where $i = 1, 2, 3$ and n = initial state, goal state from question

f) Apply Greedy Best first search to the problem with h_1 as heuristic function and $g(n)$ you defined in part-a. Show clearly computation of evaluation function and search steps show steps up to generation of one node at depth level 4 or you reach the solution which ever is early

→ Greedy Best first search

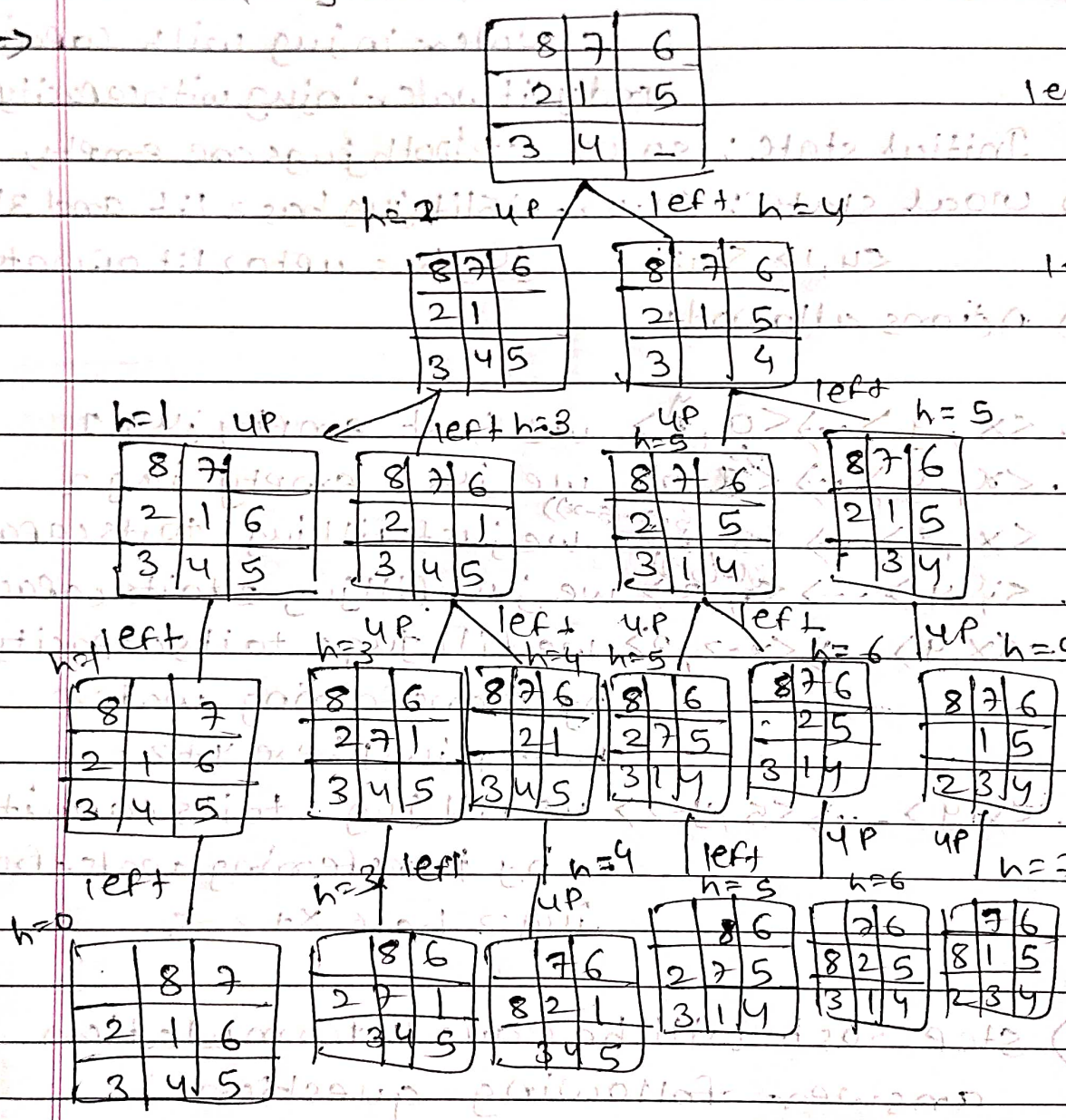


In the greedy best first search the heuristic value is less to go the solution to get the minimum node of the solution at the level 4.

PAGE No.
DATE / /

9) Apply A* search to the problem with the heuristic function and g(n) you defined in part a. Show clearly computation of evaluation function and search steps. Show step up to generation of one node at depth level 4 or you reach the solution which ever is early

→



Goal state

In the A* search the goal state which is the depth level 4 for the A* search to get the goal state.

PAGE No.
DATE / /

Q 3 consider following formulation of water jug problem:

1) state definition: <x, y> ---> indicates x lit water in jug with capacity 5 and y lit water in jug with capacity 3 lit

2) Initial state: <0, 0> ---> both jugs are empty

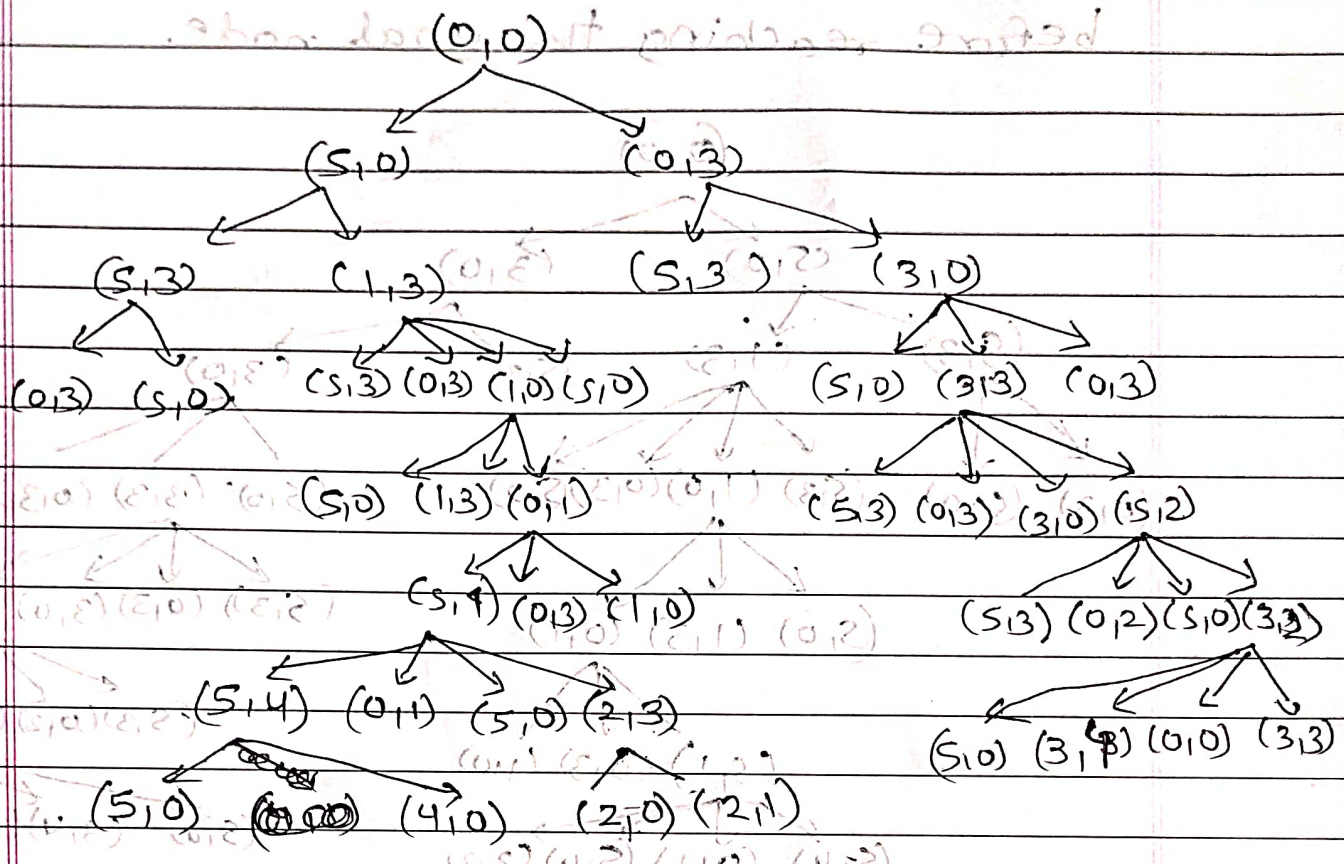
3) Goal state: <4, 0> ---> 5 lit jug has 2 lit and 3 lit
<4, 1>, <4, 2>: jug has upto 2 lit of water

4) Actions allowed:

1. <x, y> ---> <0, y> we just empty jug 1
2. <x, y> ---> <x, 0> we just empty jug 2
3. <x, y> ---> <5, y-(5-x)> we just fill jug 1 to its capacity
4. <x, y> ---> <x-(3-y), 3> we just fill jug 2 to its capacity
5. <x, y> ---> <x-z, 3> we fill jug 2 to its capacity by transfering water from jug 1 here y+z = 3
6. <x, y> ---> <5, y-z> we fill jug 1 to its capacity by transfering water from jug 2 here x+z = 5

5) step cost will be equal formulation answer following question:

a) Draw fully expanded search tree if you are doing exhaustive search up to three levels for specification given.



b) How many goal nodes are present in above search tree mention their respective depth

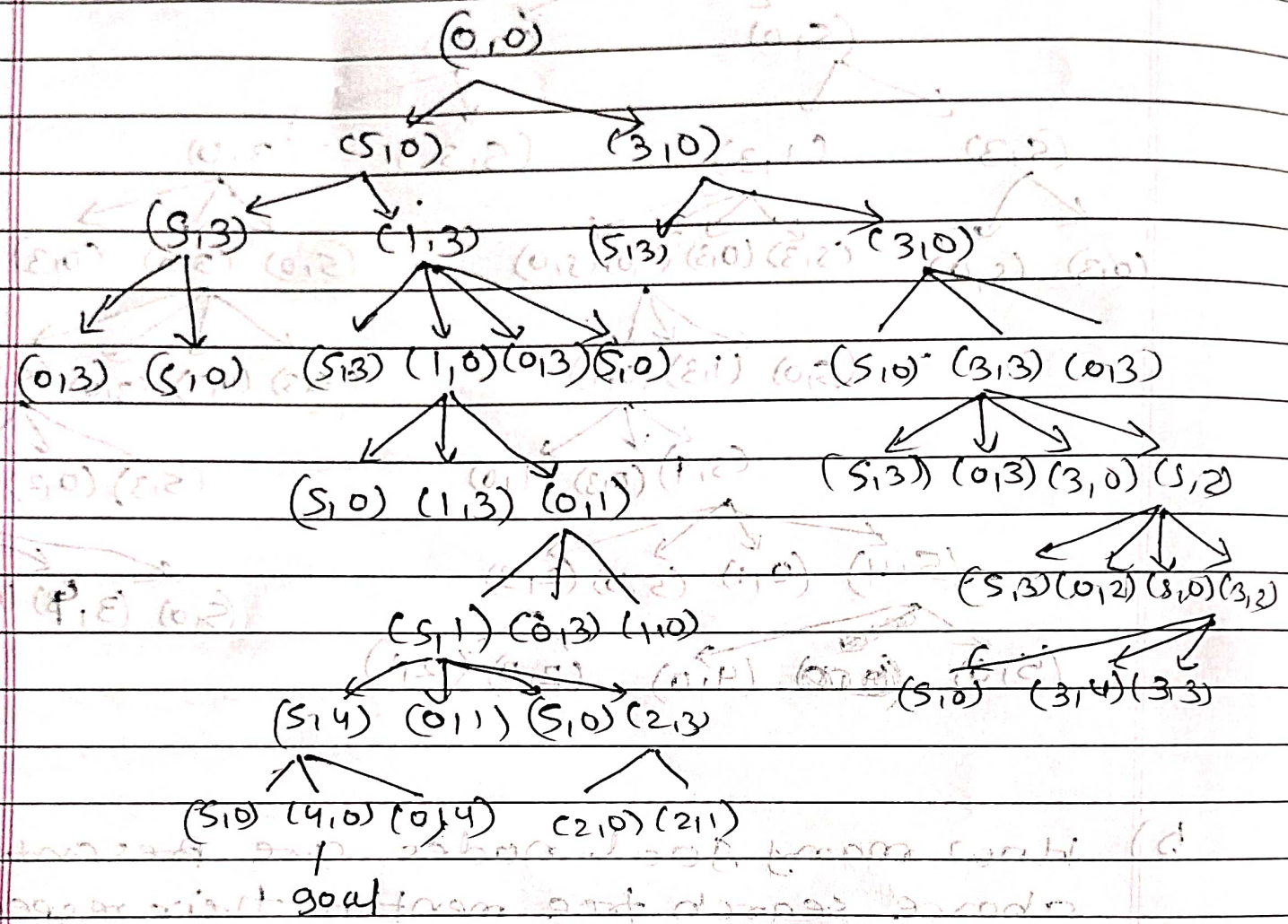
→ one (4,0)

c) what is the depth of shallowest goal node also mention are problem state for same

→ (4,0)

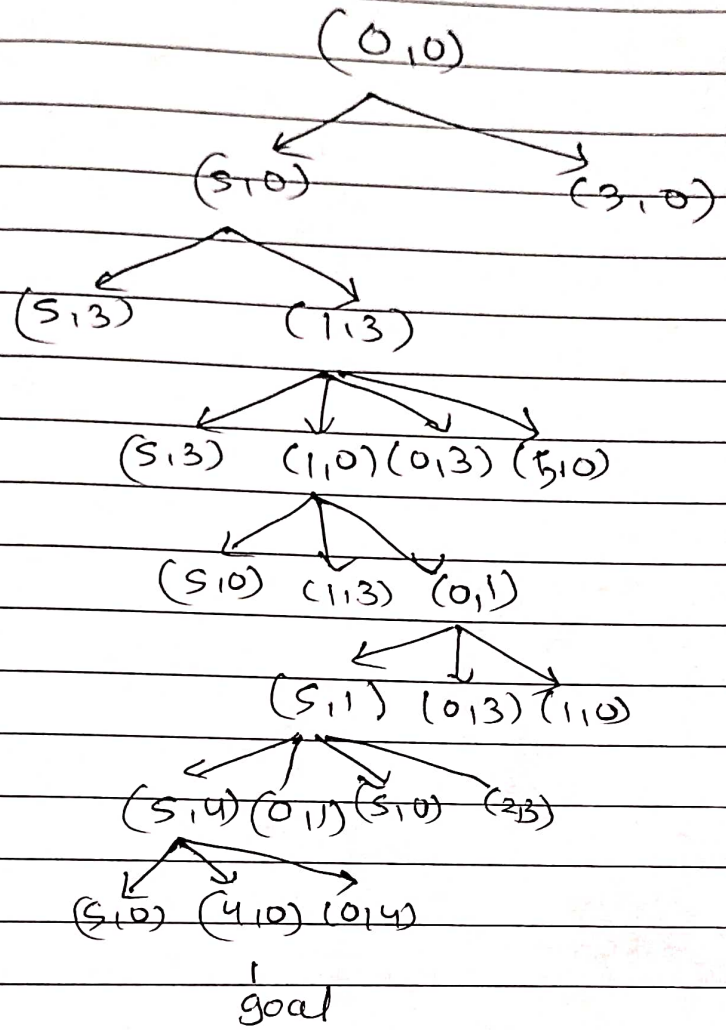
d) Apply breadth first search for above problem and find out the sequence of actions in the solution. How many nodes and in which order are visited.

before reaching the goal node.



e) Apply Depth first search for above problem and find out the sequence of action in the solution. How many nodes and in which order are visited before reaching the goal node.

→



f) what should be minimum value of depth limit set by the algorithm so that depth limited depth first search when applied to this problem gives you optimal solution. Explain how you arrive at this value.