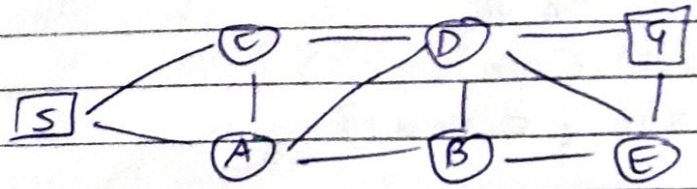
IS assignment - 1

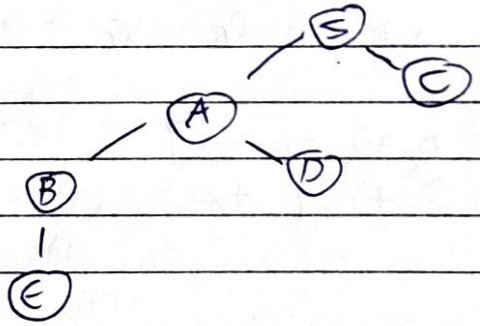
Name: Soham Walam Roll No: D 102

a)



S → A, C
A → B, C, D, S.
B → A, D, E.
C → A, D, S
D → A, B, C, E, G
E → B, D, G
G → D, E

i) DFS:



ii) Node = S
closed: (S, NULL).
MoveGen: A, C → New nodes = A, C.
New pairs: [(A, S), (C, S)].
OPEN = [(A, S), (C, S)].
closed = [S, Null].

2) Node = A.
Closed = (A,S).
MoveGen = B, G, D, S ⇒ New nodes: B, D.
New pairs. [(B,A), (D,A)].
OPEN : [(B,A), (D,A), (G,S)]
CLOSED : [(A,S), (S,NULL)]

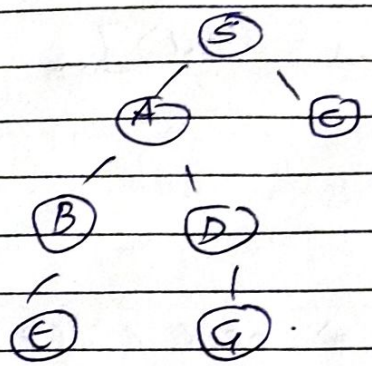
3) Node : B.
closed : (B,A).
MoveGen : A, D, E ⇒ New nodes : E.
New pairs : [(E,B)].
open : [(E,B), (D,A), (G,S)].
closed : [(B,A), (A,S), (S,NULL)].

4) Node : E.
closed : (E,B).
MoveGen : B, D, G ⇒ New node : G
New pairs : [(G,E)].
open : [(G,E), (D,A), (G,S)].
closed : [(E,B), (B,A), (A,S), (S,NULL)].

5) Node : G ← goal.
OPEN : [(G,E), (D,A), (G,S)].
CLOSED : [(E,B), (B,A), (A,S), (S,NULL)].

→ Path : S : A : B : E : G.

BFS:



1) Node : S.
closed : (S, Null).
MoveGen : A, E ⇒ new node : A, E.
new pairs : [(A, S), (E, S)].
open : [(A, S), (E, S)]
closed : [(S, Null)].

2) Node : A
closed : (A, S).
MoveGen : B, C, D, S ⇒ new nodes : B, D.
new pairs : [(B, A), (D, A)].
Open ~~closed~~ : [(B, A), (D, A), (E, S)].
closed : [(A, S), (S, Null)]

3) Node = E
closed : (E, S).
MoveGen : A, D, S ⇒ new nodes : —

4) Node = B
closed : (B, A)
MoveGen : A, D, E ⇒ new nodes : E.
new pairs : ((E, B)).

open : [(E, B), (D, A)].
closed : [(B, A), (A, S), (S, Null)].
(G, D).

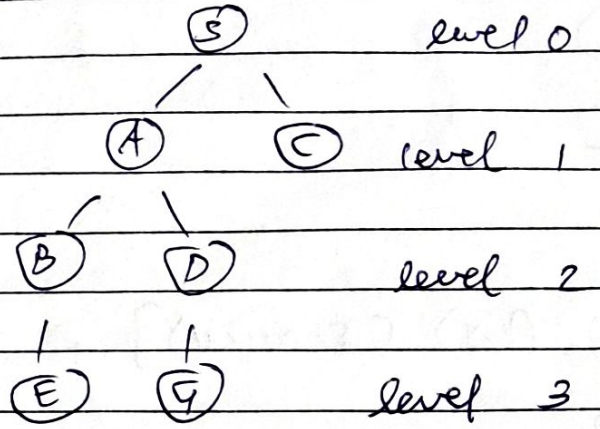
5) Node : D.
closed : (D, A).
Move Gen : A, B, C, E, G ⇒ new node : G.
new pairs : (G, D).
open : [(G, D), (E, B)]
closed : [(D, A), (B, A), (G, S), (A, S), (S, Null)].

6) Node : E,
closed : (E, B)
move gen : B, D, G ⇒ new node : -

7) Node : G ← goal.
open (G, D).
closed : [(E, B), (D, A), (B, D), (G, S), (A, S), (S, Null)].

→ Path : S : A : C : B : D : E : G.

* DFID:



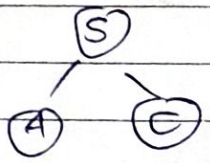
At level 0.

1) Node = S.
closed: [S, Null)
Move Gen: -
new nodes: -
closed : [S, Null].
open: []

2) At level 1,

Node: S
closed: [S, Null).
MoveGen: A, C.
New nodes A, C.
New pairs: [(A,S), (C,S)].
open : [(A,S), (C,S)]
closed (S, Null).

3) Node : C
closed : (C,S).
Move gen : -
New nodes : -
new pairs : -
Open : [ ].
closed : [(C,S),(A,S),(S,Null)].



G not found increasing depth.

At level 2.

1) Node : S
closed : (S,Null).
Move gen : A/C.
New nodes : A,C.
New pairs : [(A,S),(C,S)].
Open : [(A,S),(C,S)].
closed : [S,Null].

2) Node : A.
closed . (A,S)
MoveGen : B,C,D,S.
new nodes : B,D.
new pairs : [(B,A,)(D,A)].
Open : [(B,A),(D,A),(C,S)].
closed : [(A,S),(S,Null)].

3) Node = C
closed : (C,S).
Move Gen : A, D, S.
new nodes : -
new pairs : -
open : [(B,A), (D,A)].
closed : [(C,S), (A,S), (S, Null)]

4) Node = B
→ closed : (B,A)
Move Gen : —
new node : —
new pairs : —
open : [(D,A)].
closed : [(B,A), (C,S), (A,S), (S, Null)].

5) Node = D.
closed : (D,A)
Move Gen : —
→ new nodes : —
new pairs : —
open : []
closed : [(D,A), (B,A), (C,S), (A,S), (S, Null)]

Goal not found. increasing depth.

At level 3, upto Node C same as level 2.

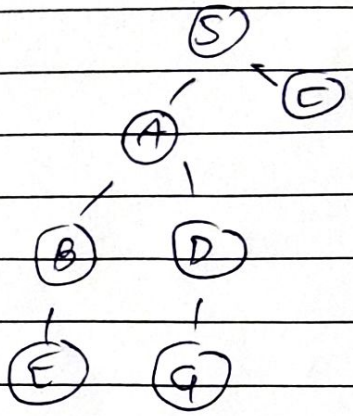
4) Node = B.
closed = (B, A).
Move gen = A, D, E.
new nodes = E.
new pairs = [(E, B)].
open = [(E, B), (D, A)].
closed = [(B, A), (C, S), (A, S), (S, NULL)].

5) Node = D.
closed = (D, A)
Move gen = A, B, C, E, G.
new nodes = G
new pairs = [(G, D)].
open = [(G, D), (E, B)].
closed = [(D, A), (B, A), (C, S), (A, S), (S, NULL)].

6) Node = E.
closed = (E, B).
Move gen = —
new nodes = —
new pairs = —
open = [(G, D)]
closed = [(E, B), (D, A), (B, A), (C, S), (A, S), (S, NULL)].

7) node = G
goal = G
open = ((G, D)).
closed = ((E, B), (D, A), (B, A), (C, S), (A, S), (S, NULL)].

Path : S : A : B : E : D : G



We reached soln at level 3.