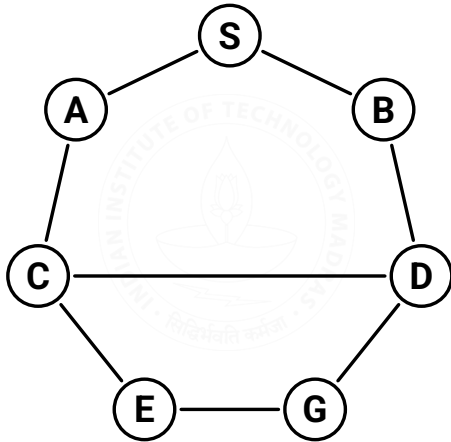


# Lecture Example 3

## Depth First Iterative Deepening (DFID-N)

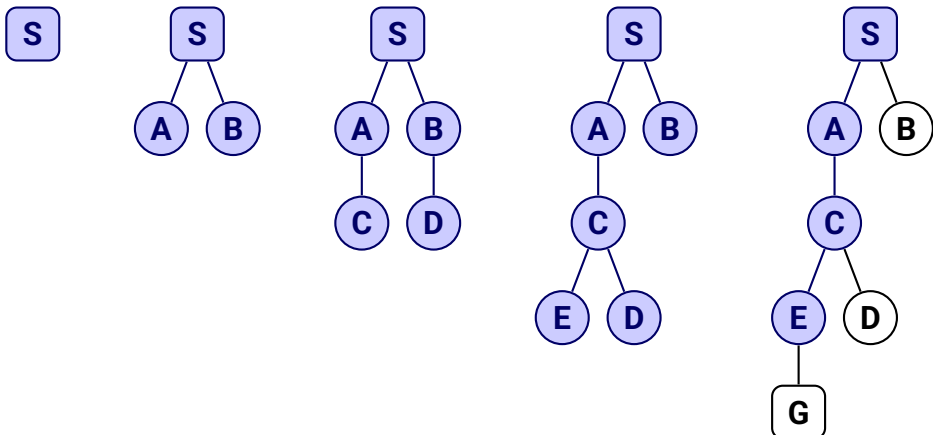
Prepared by S. Baskaran

### State Space



X	MoveGen(X)
S	[A, B]
A	[S, C]
B	[S, D]
C	[E, D, A]
D	[B, C, G]
E	[C, G]
G	[D, E]

### DFID-N Search Trees (d=0,1,2,3,4)



# DFID-N Solution

The solution provided below is based on the DFID-N algorithm published in the Week 2 Notes.

OPEN and CLOSED carry triples: (NODE,PARENT,DEPTH)

\*\*\*[DEPTH BOUND = 0]\*\*\*

OPEN (S,null,0):[]  
CLOSED []

1.  
NODE S  
close (S,null,0)

OPEN []  
CLOSED (S,null,0):[]

\*\*\*[DEPTH BOUND = 1]\*\*\*

OPEN (S,null,0):[]  
CLOSED []

1.  
NODE S  
close (S,null,0)  
moveGen A:B:[]  
newNodes A:B:[]  
newPairs (A,S,1):(B,S,1):[]

OPEN (A,S,1):(B,S,1):[]  
CLOSED (S,null,0):[]

2.  
NODE A  
close (A,S,1)

OPEN (B,S,1):[]  
CLOSED (A,S,1):(S,null,0):[]

3.  
NODE B  
close (B,S,1)

OPEN []  
CLOSED (B,S,1):(A,S,1):(S,null,0):[]

\*\*\*[DEPTH BOUND = 2]\*\*\*

OPEN (S,null,0):[]  
CLOSED []

1.  
NODE S  
close (S,null,0)  
moveGen A:B:[]  
newNodes A:B:[]  
newPairs (A,S,1):(B,S,1):[]

OPEN (A,S,1):(B,S,1):[]  
CLOSED (S,null,0):[]

2.  
NODE A  
close (A,S,1)  
moveGen S:C:[]  
newNodes C:[]  
newPairs (C,A,2):[]

OPEN (C,A,2):(B,S,1):[]  
CLOSED (A,S,1):(S,null,0):[]

3.  
NODE C  
close (C,A,2)

OPEN (B,S,1):[]  
CLOSED (C,A,2):(A,S,1):(S,null,0):[]

4.  
NODE B  
close (B,S,1)  
moveGen S:D:[]  
newNodes D:[]  
newPairs (D,B,2):[]

OPEN (D,B,2):[]  
CLOSED (B,S,1):(C,A,2):(A,S,1):(S,null,0):[]

5.  
NODE D  
close (D,B,2)

OPEN []  
CLOSED (D,B,2):(B,S,1):(C,A,2):(A,S,1):(S,null,0):[]

\*\*\*[DEPTH BOUND = 3]\*\*\*

OPEN (S,null,0):[]  
CLOSED []

1.  
NODE S  
close (S,null,0)  
moveGen A:B:[]  
newNodes A:B:[]  
newPairs (A,S,1):(B,S,1):[]

OPEN (A,S,1):(B,S,1):[]  
CLOSED (S,null,0):[]

2.  
NODE A  
close (A,S,1)  
moveGen S:C:[]  
newNodes C:[]  
newPairs (C,A,2):[]

OPEN (C,A,2):(B,S,1):[]  
CLOSED (A,S,1):(S,null,0):[]

3.  
NODE C  
close (C,A,2)  
moveGen E:D:A:[]  
newNodes E:D:[]  
newPairs (E,C,3):(D,C,3):[]

OPEN (E,C,3):(D,C,3):(B,S,1):[]  
CLOSED (C,A,2):(A,S,1):(S,null,0):[]

4.  
NODE E  
close (E,C,3)

OPEN (D,C,3):(B,S,1):[]  
CLOSED (E,C,3):(C,A,2):(A,S,1):(S,null,0):[]

5.  
NODE D  
close (D,C,3)

OPEN (B,S,1):[]  
CLOSED (D,C,3):(E,C,3):(C,A,2):(A,S,1):(S,null,0):[]

6.  
NODE B  
close (B,S,1)  
moveGen S:D:[]  
newNodes []  
newPairs []

OPEN []  
CLOSED (B,S,1):(D,C,3):(E,C,3):(C,A,2):(A,S,1):(S,null,0):[]

\*\*\*[DEPTH BOUND = 4]\*\*\*

OPEN (S,null,0):[]  
CLOSED []

1.  
NODE S  
close (S,null,0)  
moveGen A:B:[]  
newNodes A:B:[]  
newPairs (A,S,1):(B,S,1):[]

OPEN (A,S,1):(B,S,1):[]  
CLOSED (S,null,0):[]

2.  
NODE A  
close (A,S,1)  
moveGen S:C:[]  
newNodes C:[]  
newPairs (C,A,2):[]

OPEN (C,A,2):(B,S,1):[]  
CLOSED (A,S,1):(S,null,0):[]

3.  
NODE C  
close (C,A,2)  
moveGen E:D:A:[]  
newNodes E:D:[]  
newPairs (E,C,3):(D,C,3):[]

OPEN (E,C,3):(D,C,3):(B,S,1):[]  
CLOSED (C,A,2):(A,S,1):(S,null,0):[]

4.  
NODE E  
close (E,C,3)  
moveGen C:G:[]  
newNodes G:[]  
newPairs (G,E,4):[]

OPEN (G,E,4):(D,C,3):(B,S,1):[]  
CLOSED (E,C,3):(C,A,2):(A,S,1):(S,null,0):[]

5.  
NODE G  
GOAL G

OPEN (G,E,4):(D,C,3):(B,S,1):[]  
CLOSED (E,C,3):(C,A,2):(A,S,1):(S,null,0):[]

PATH S:A:C:E:G:[]