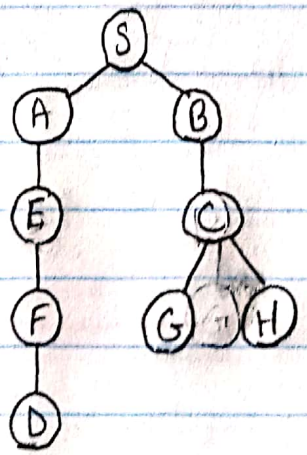
CS 435 - Project 2

1a. S, A, B, C, E, F, G, K, L, D

1b. adjacency matrix:

	A	B	C	D	E	F	G	K	L	S
A	0	1	1	0	1	0	0	0	0	1
B	1	0	0	0	1	0	0	0	0	0
C	1	0	0	0	0	0	0	0	0	0
D	0	0	0	0	0	0	0	0	1	0
E	1	1	0	0	0	1	0	0	0	0
F	0	0	0	0	1	0	1	0	0	0
G	0	0	0	0	0	1	0	1	0	0
K	0	0	0	0	0	0	1	0	1	0
L	0	0	0	1	0	0	0	1	0	0
S	1	0	0	0	0	0	0	0	0	0

1c.

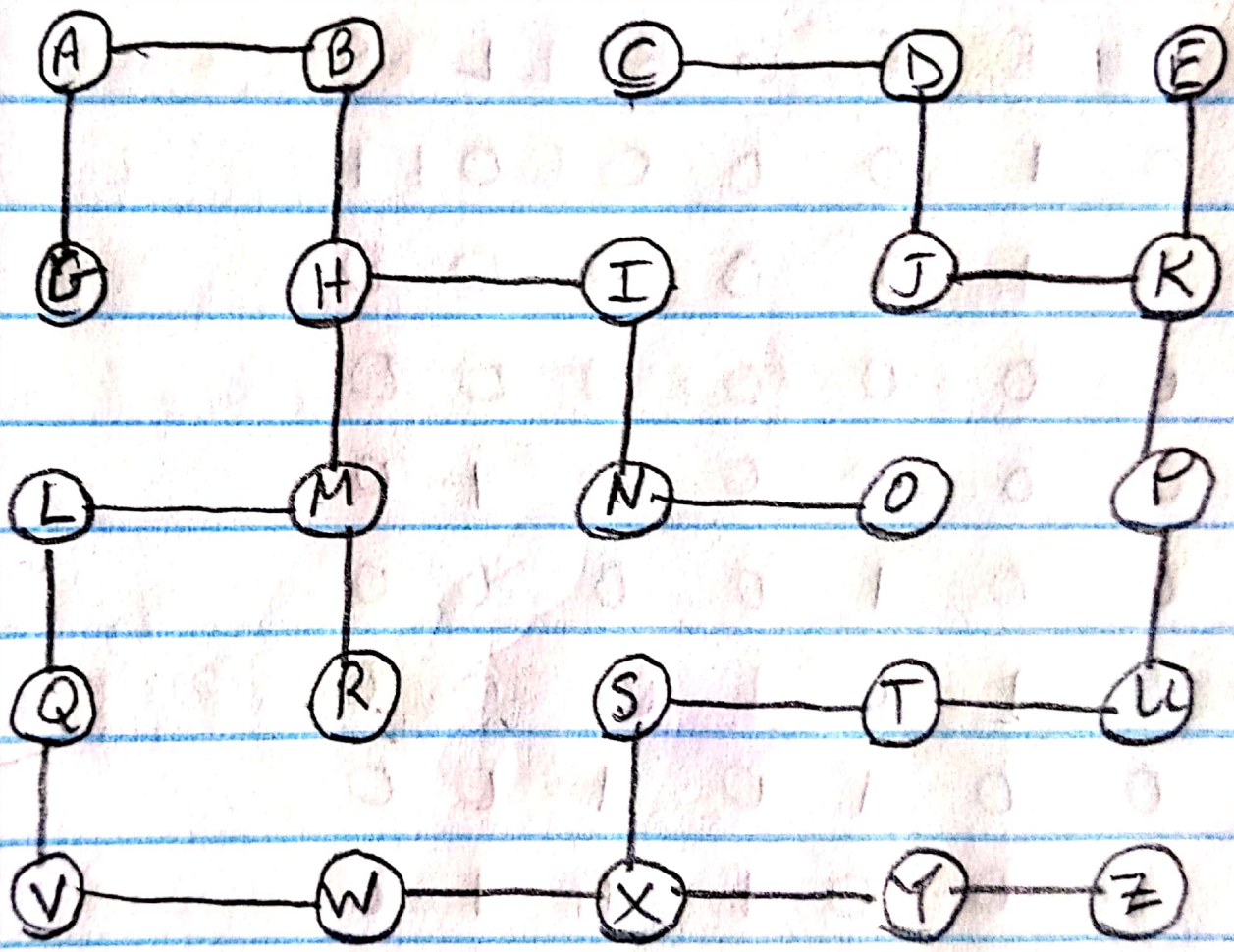


2a. 25 nodes

2b. an undirected edge exists between two nodes if there is no wall between the two nodes

2c. undirected, acyclic, connected, unweighted

ad.



Devan Patel
CS 435
Proj 2 - Part 1

3h) Recieved a stack overflow error with 10,000 nodes. Works perfectly fine with 100 nodes. The stack overflow is caused by the implicit space used by recursion. A new stack frame is pushed onto the stack for each recursive call. The more nodes we have, the more nodes we will have in the queue. The recursive call happens for each node in the queue and hence, we have more recursive calls and more implicit space used.

3i) This solution does not result in any issues because it does not make any recursive calls. Basically, it uses no additional implicit space. Additional space used is on the heap. The recursive solution might run into problems as explained in 3h.