Tutorial - 5

Kidlima Gandhi
G
08

Q1

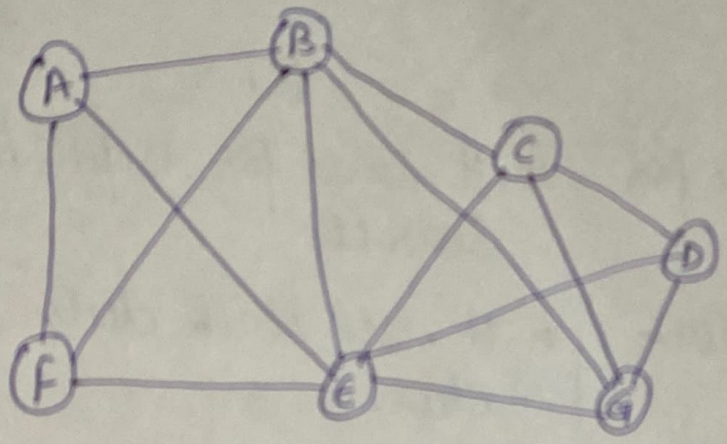
BFS	DFS
* It stand for Breadth for search	* It stand for Depth first search
* It uses queue data structure	* It uses stack data structure
* It is more suitable for searching	* It is more suitable when there are solution away from source.
* Here sibling are visited before children	* Here children are visited before sibling
* There is no concept of backtracking	* It is recursive algo that use backtracking

Applications

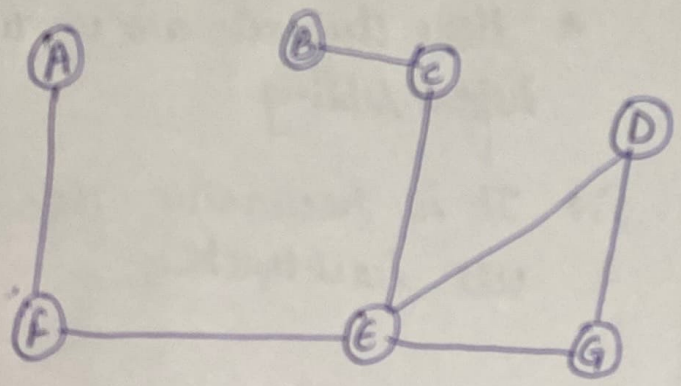
- BFS → Bipartite graph & shortest path
- DFS → acyclic graph, topological order

Q3

Dequen graph is a graph in which no of edge is close to maximal no of edges.



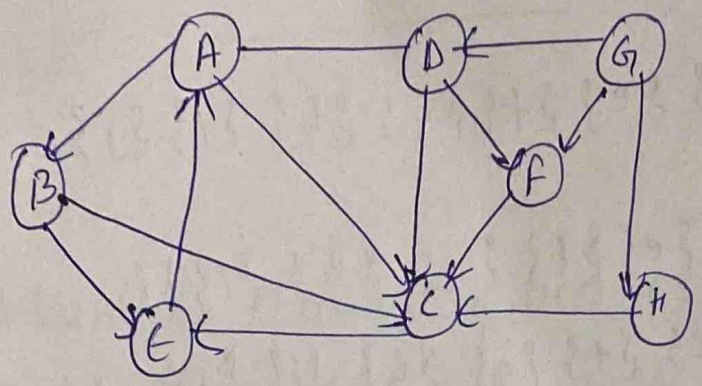
Dense Graph
(many edges b/w nodes)



Sparse graphs (few edges b/w nodes)

* For sparse graph it is performed to use Adjacency list
* For dense graph it is preferred to use adjacency matrix

Q6



BFS

Child	G	H	D	F	C	E	A	B
Parent		G	G	G	H	C	E	A

Path → G → H → C → E → A → B

DFS

Nodes visited	Stack
G	G
D	F
H	C
F	E
C	A
F	B
A	
B	

Path → G → f → C → E → A → B

Q7

V = {a} {b} {c} {d} {e} {f} {g} {h} {i} {j}

(a, b) {a, b} {c} {d} {e} {f} {g} {h} {i} {j}

(a, c) {a, b, c} {d} {e} {f} {g} {h} {i} {j}

[b, c) {a, b, c} {d} {e} {f} {g} {h} {i} {j}

(e, g) {a, b, c, d) {e, f, g} {h} {i} {j}

[h, i} {a, b c d} {e, f, g} {h, i} {j}

No of connected components = 3