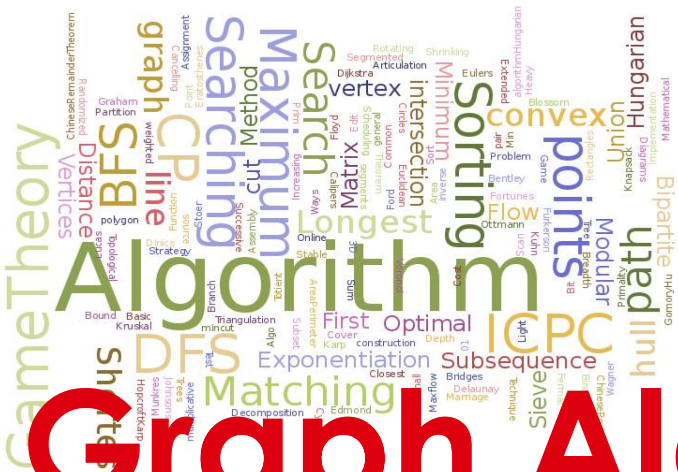
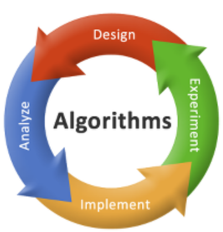


Graph Algorithms - An Overview

Course: Algorithms

Faculty: Dr. Rajendra Prasath



Autumn 2018

Graph Algorithms – An Overview – BFS and DFS

This lecture covers the overview of graph algorithms with two Tree based algorithms: Breadth First Search and Depth First Search. We provide illustrations and the complexity analysis of these two graph algorithms

Recap: Search Algorithms

- Given: A collection C of elements
- Task: Search for the element x

- **Existence:**
  - Does C contain a target element x?
  - Response: Yes (exists) / No (does not exist)
- **Retrieval:**
  - Give your roll number and get all details?
  - Search anything in Google – Scalable searches
- **Associate Look up:**
  - Information Associated with the key x

Recap: Search Algorithms

Different Search Algorithms:

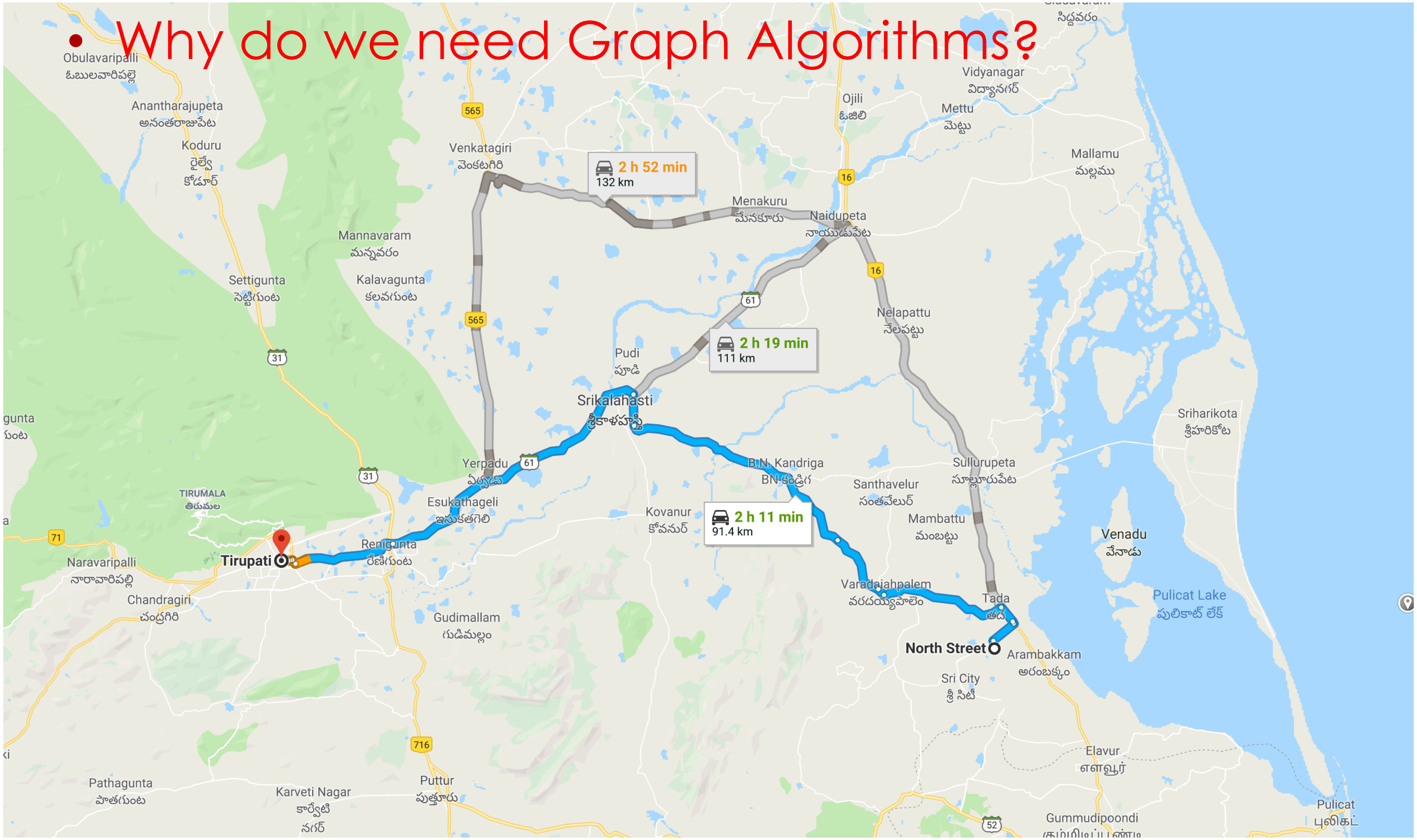
- Linear Search Algorithm
- Binary Search Algorithm

- Hash Based Search Algorithm
- Binary Search Tree Algorithm

- Complexity analysis and the worst case running time of each algorithm

Graph Algorithms

- Why do we need Graph Algorithms?



Graph Algorithms

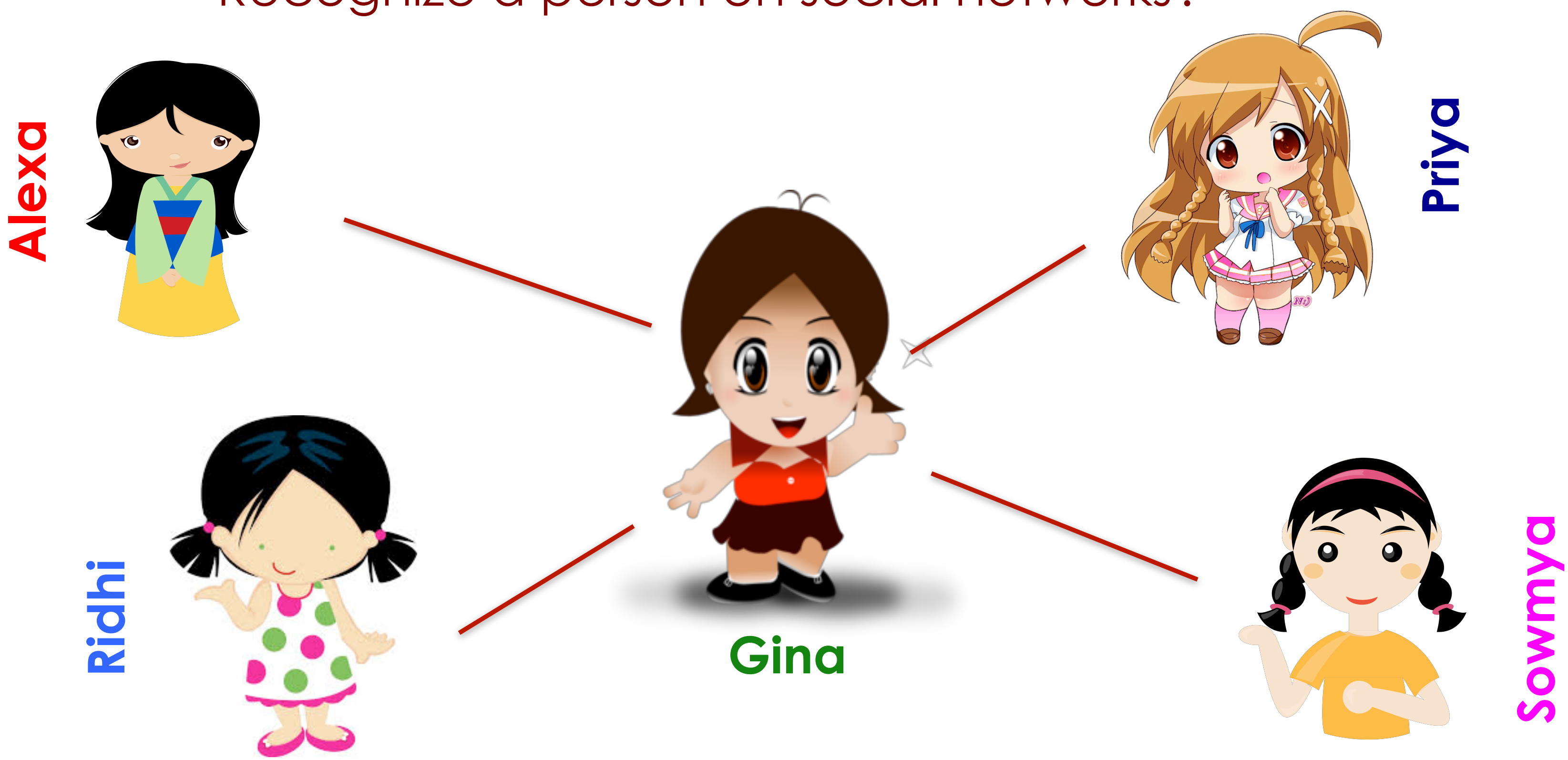
- Maps (Google, Bing, Wiki and so on)
  - Is this solving the Shortest Path Problem?
    - Shortest Path: A navigation service to find the optimal route and estimate time to destination
  - Finding the fastest routes (minimize travel time!)
    - How quickly one can reach the destination from the given source
  - Finding alternative routes (diversions … !!)
    - Diversion may alter the course of travel and could bring up more constaints

Graph Algorithms

- Social Network
  - Assortativity (preference of nodes to get attached to others that are similar in some way)
    - Rich gets Richer
    - How many of you like Sachin Tendulkar? Why?
      - For his Cricket?
      - For his Politeness / Humble Nature?
      - For his ODI Records?
    - If you are a tennis lover, which Indian Player will you follow - Leander Paes / Sania Mirza ??
      - Either in social network or through his advertisement or by some other means??

Graph Algorithms - Example

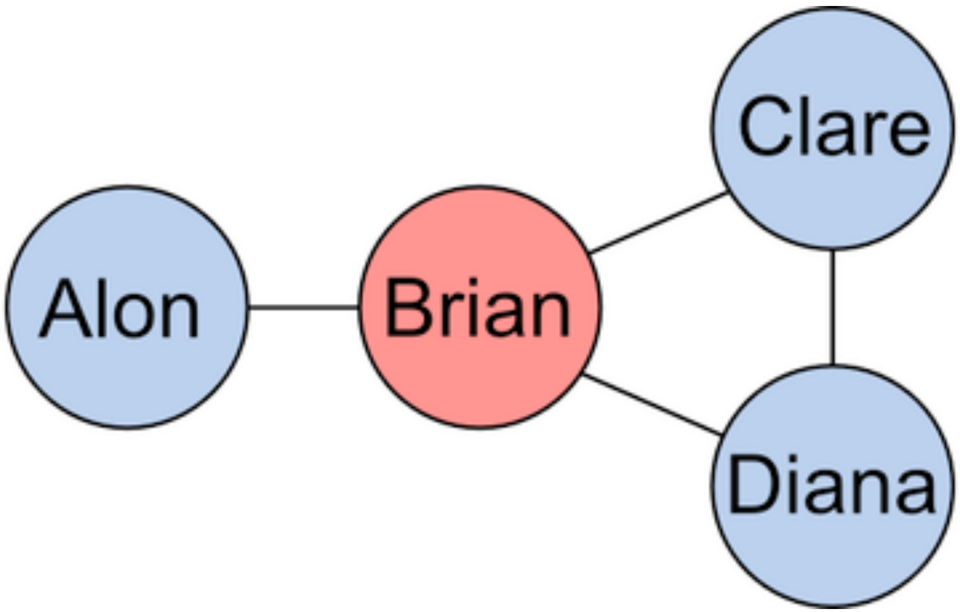
- Social Networks
  - Recognize a person on social networks?



- Friend of a Friend is also a **Friend (Mutual Friend)**

Mutual Friends

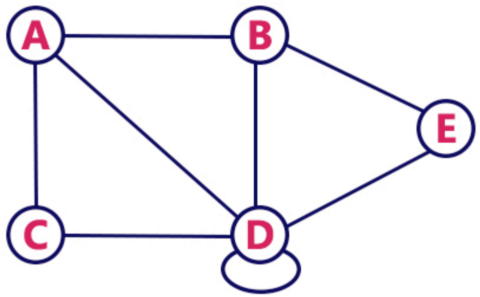
- Social Network
  - How do you recognize a person on a social network?



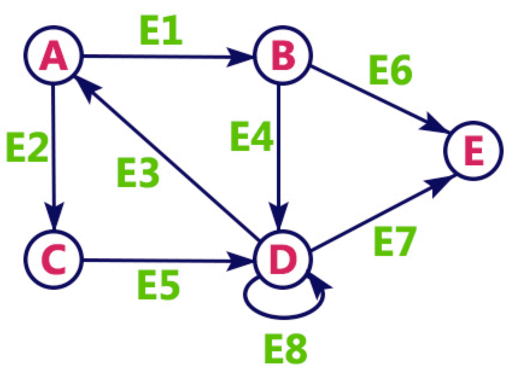
  - Persons – Connections and their relations

Graph Representations

- Examples



	A	B	C	D	E
A	0	1	1	1	0
B	1	0	0	1	1
C	1	0	0	1	0
D	1	1	1	1	1
E	0	1	0	1	0



	E1	E2	E3	E4	E5	E6	E7	E8
A	1	1	-1	0	0	0	0	0
B	-1	0	0	1	0	1	0	0
C	0	-1	0	0	1	0	0	0
D	0	0	1	-1	-1	0	1	1
E	0	0	0	0	0	-1	-1	0

Graph Algorithms

- Problems could be better represented and solved using Graph Algorithms
  - We focus on a traversal problem:
    - Given G=(V, E) and a vertex v
    - Find all w in V such that w connects v
- Two Graph Traversals:
  - Breadth First Search (BFS)
  - Depth First Search (DFS)
    - Both algorithms will result in getting the connected components
    - Spanning Trees

Graph (Tree) Traversal

- Types of Traversals
- Pre-order

- In-order

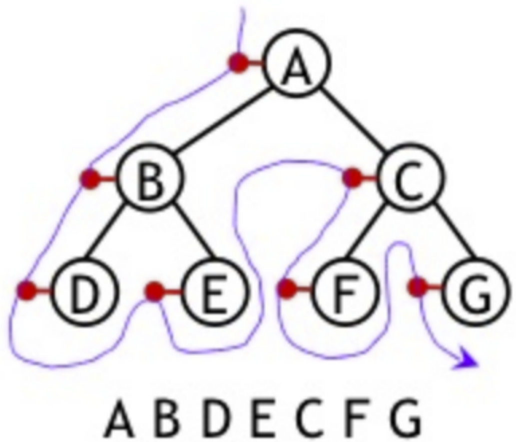
- Post order

- Level Ordering

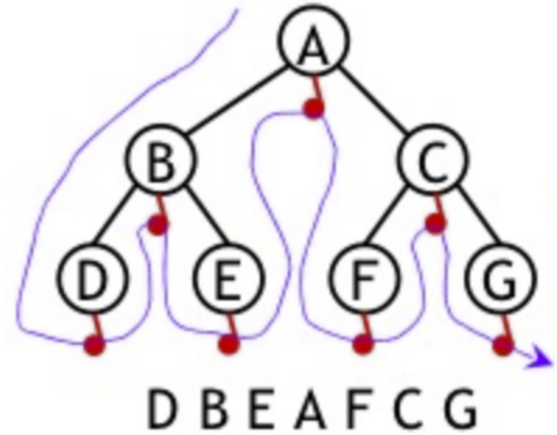
Tree Traversal - Examples

- A few examples

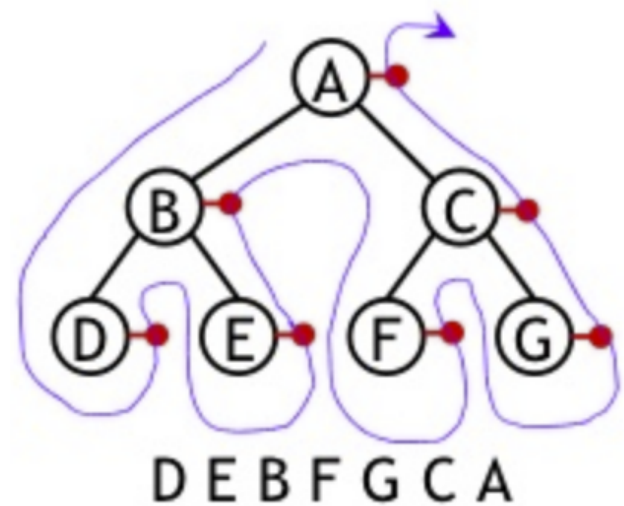
Pre-order



In-order

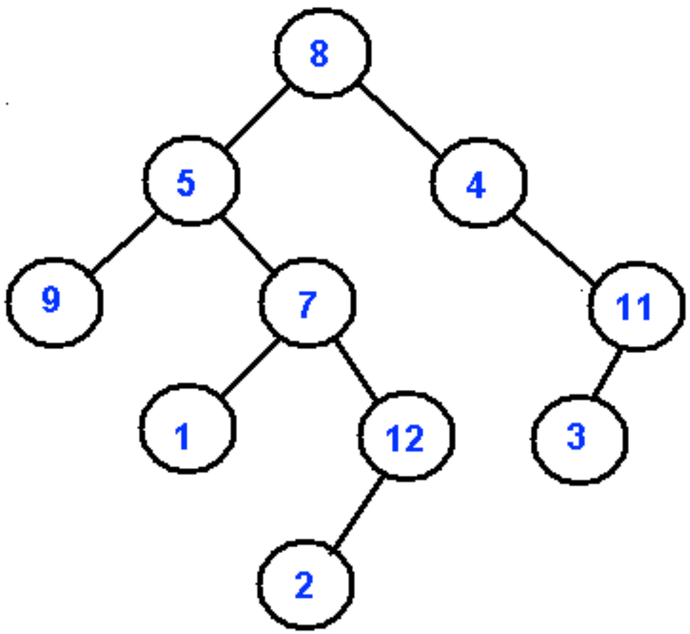


Post-order



Tree Traversal - Illustration

- Look at the graph (tree)



Pre-order
8, 5, 9, 7, 1, 12, 2, 4, 11, 3

In-order
9, 5, 1, 7, 2, 12, 8, 4, 3, 11

Post-order
9, 1, 2, 12, 7, 5, 3, 11, 4, 8

Level-order
8, 5, 4, 9, 7, 11, 1, 12, 3, 2

Breadth First Search (BFS)

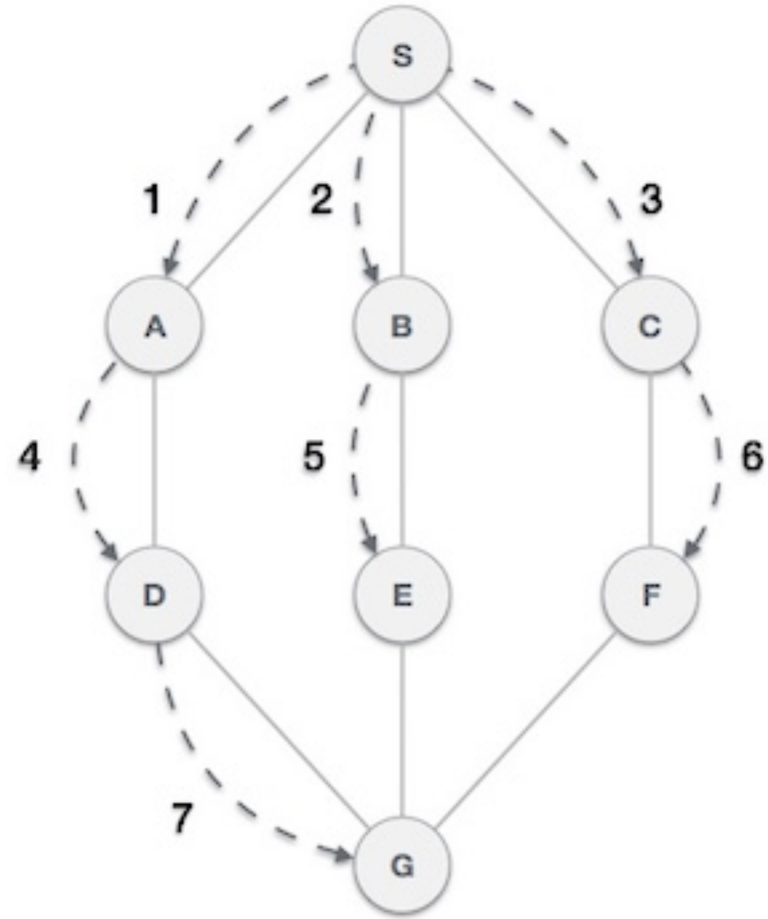
- What is Breadth First Search?
  - Level order traversal
    - Look at all possible paths at the same depth before you go at a deeper level
    - Back up as far as possible when you reach a "dead end" (i.e., next vertex has been "marked" or there is no next vertex)

BFS - Algorithm

- Basic Idea:
  - Visit all nodes reachable from that node
  - After visiting all nodes in the immediate next level move to next level
  - Start again to visit all nodes reachable from that node
  - Repeat the same until all nodes are visited
  - Addition of a node should not form a cycle

BFS - Illustration

- Look at the Graph
- Breadthward Search
- Visit all nodes in the next level and then move to another level to start the search for



BFS - Applications

- Shortest Path and Minimum Spanning Tree
- Peer to Peer Networks
- Crawlers in Search Engines
- Social Networking Websites
  - Find people within a given distance 'k' from a person

- GPS Navigation systems
  - Find all neighboring locations.

- Broadcasting in Network
- Garbage Collection
- Cycle detection in undirected graph
- Path Finding

BFS – Complexity

- Given: Graph G = (V, E)
- Best Case:
  - O(V+E)
- Average Case:
  - O(V+E)
- Worst Case:
  - O(V+E)
- Data Structures:
  - Arrays, Graphs, Recursive and Queue

Depth First Search - Overview

- Depth First
  - Pre Order Traversal

- Basic Idea:
  - Start with any node
  - Follow a depth-ward motion
  - Stack can be used to remember to get the next vertex to start a search

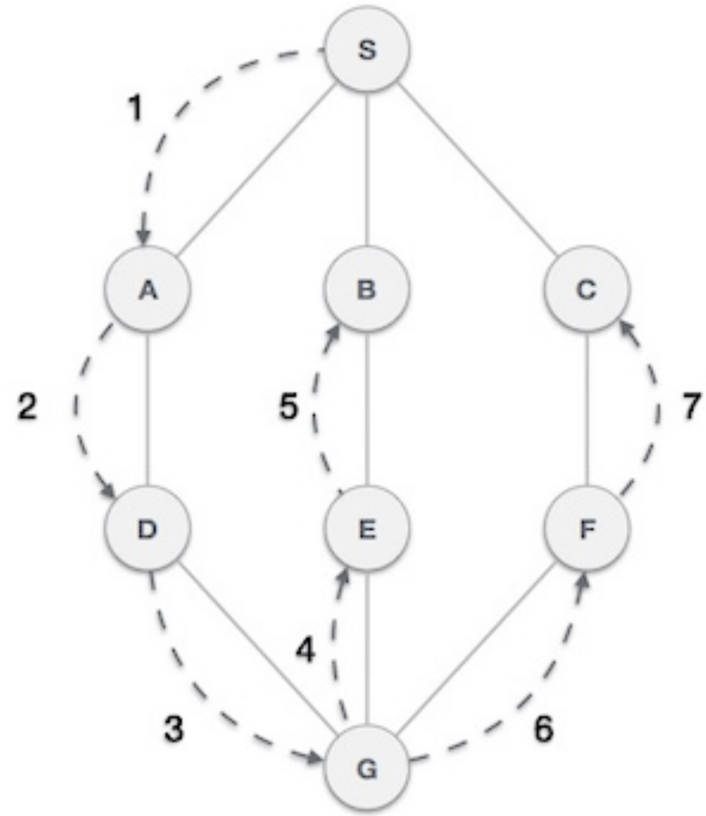
DFS - Algorithm

- Basic Idea:
  - Start with a specific node
  - Travel as far as you can down a path
  - Back up as little as possible when you reach a "dead end" (i.e., next vertex has been "marked" or there is no next vertex)
  - DFS can be implemented efficiently using a stack

DFS - Illustration

- Look at this example

- Ensure that addition of a new node **does not form a cycle**



DFS - Applications

- Path Finding
- Detecting Cycles in a graph
- Test Bipartite graphs
  - Use of Two colors
- Find Strongly connected components
- Solving puzzles like Mazes

DFS – Complexity

- Given: Graph G = (V, E)
- Best Case:
  - O(V+E)
- Average Case:
  - O(V+E)
- Worst Case:
  - O(V+E)
- Data Structures:
  - Arrays, Graphs, and Queue

Help among Yourselves?

- **Perspective Students** (having CGPA above 8.5 and above)
- **Promising Students** (having CGPA above 6.5 and less than 8.5)

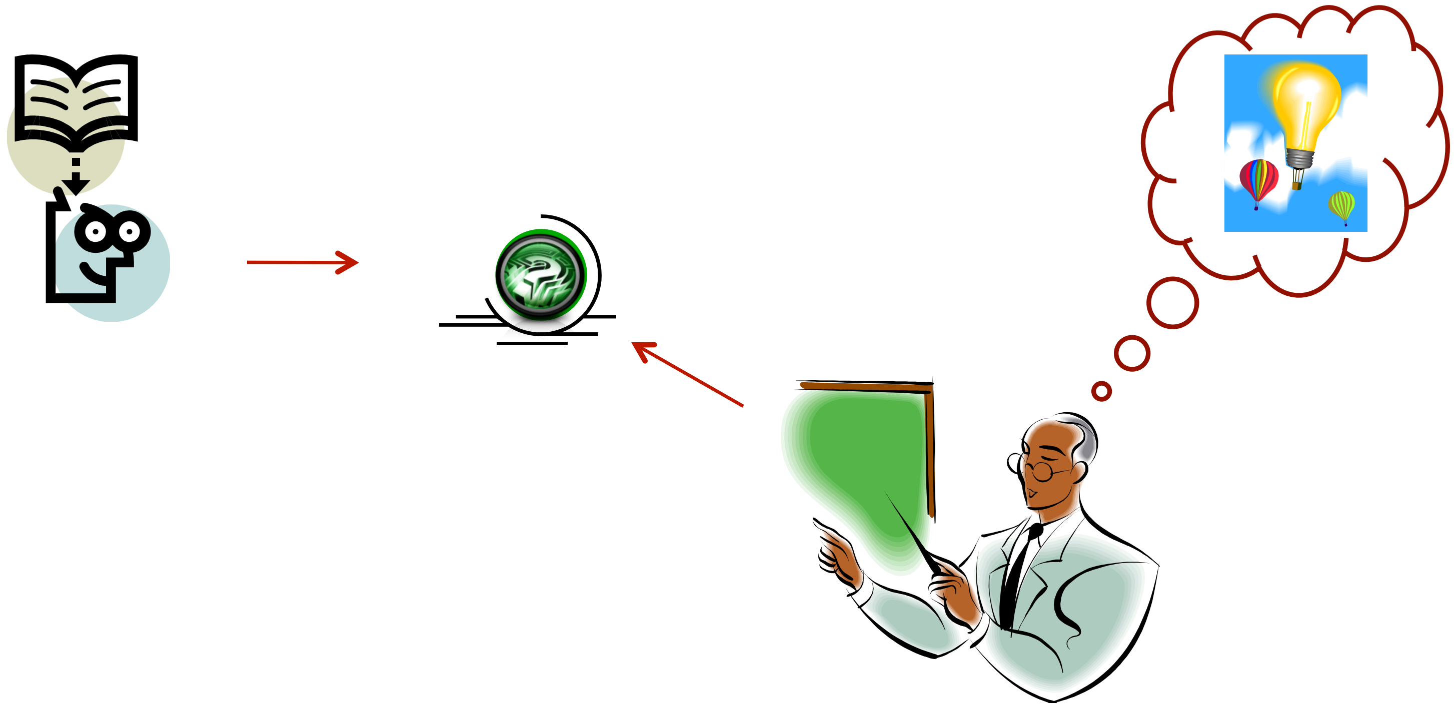
- **Needy Students** (having CGPA less than 6.5)
  - Can the above group help these students? (Your work will also be rewarded)

- You may grow a culture of **collaborative learning** by helping the needy students

Assistance

- You may post your questions to me at any time
- You may meet me in person on available time or with an appointment
- TA s would assist you to clear your doubts.
- You may leave me an email any time (email is the best way to reach me faster)

Thanks …



… Questions ???