Session 6

Data structures

Graphs and its traversal algorithms

Prepared by/ Elzahraa Alaa Tag Eldein

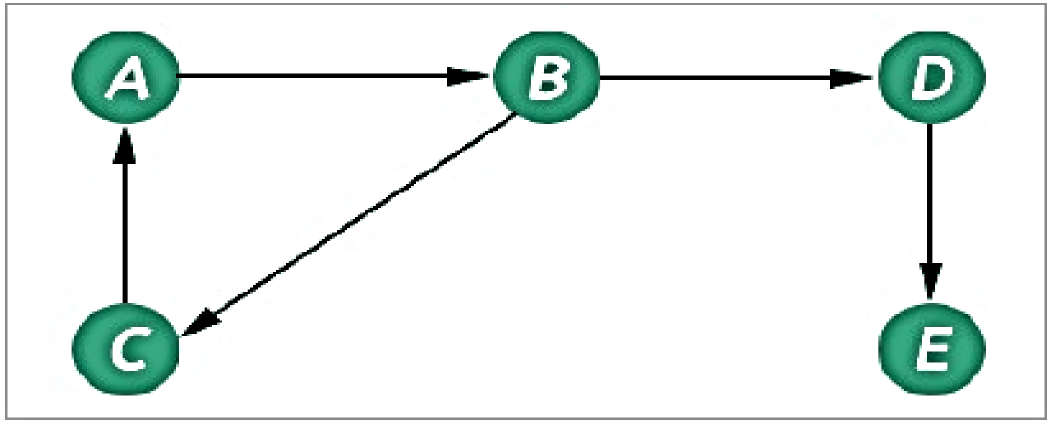
April 10, 2023

Task 1

mention traversal algorithms of graph

In this section we will see what is a graph data structure, and the traversal algorithms of it.

The graph is one non-linear data structure. That is consists of some nodes and their connected edges. The edges may be director or undirected. This graph can be represented as G(V, E). The following graph can be represented as G({A, B, C, D, E}, {(A, B), (B, D), (D, E), (B, C), (C, A)})



The graph has two types of traversal algorithms. These are called the Breadth First Search and Depth First Search.

Breadth First Search (BFS)

The Breadth First Search (BFS) traversal is an algorithm, which is used to visit all of the nodes of a given graph. In this traversal algorithm one node is selected and then all of the adjacent nodes are visited one by one. After completing all of the adjacent vertices, it moves further to check another vertices and checks its adjacent vertices again.

Algorithm

```
bfs(vertices, start)
Input: The list of vertices, and the start vertex.
Output: Traverse all of the nodes, if the graph is connected.
Begin
   define an empty queue que
   at first mark all nodes status as unvisited
   add the start vertex into the que
   while que is not empty, do
      delete item from que and set to u
      display the vertex u
      for all vertices 1 adjacent with u, do
         if vertices[i] is unvisited, then
            mark vertices[i] as temporarily visited
            add v into the queue
         mark
      done
      mark u as completely visited
   done
End
```

Depth First Search (DFS)

The Depth First Search (DFS) is a graph traversal algorithm. In this algorithm one starting vertex is given, and when an adjacent vertex is found, it moves to that adjacent vertex first and try to traverse in the same manner.

Algorithm

```
dfs(vertices, start)
Input: The list of all vertices, and the start node.
Output: Traverse all nodes in the graph.
Begin
   initially make the state to unvisited for all nodes
   push start into the stack
   while stack is not empty, do
      pop element from stack and set to u
      display the node u
      if u is not visited, then
         mark u as visited
         for all nodes i connected to u, do
            if ith vertex is unvisited, then
               push ith vertex into the stack
               mark ith vertex as visited
         done
   done
End
```

Difference between BFS and DFS

Both BFS and DFS are types of graph traversal algorithms, but they are different from each other. BFS or Breadth First Search starts from the top node in the graph and travels down until it reaches the root node. On the other hand, DFS or Depth First Search starts from the top node and follows a path to reaches the end node of the path.

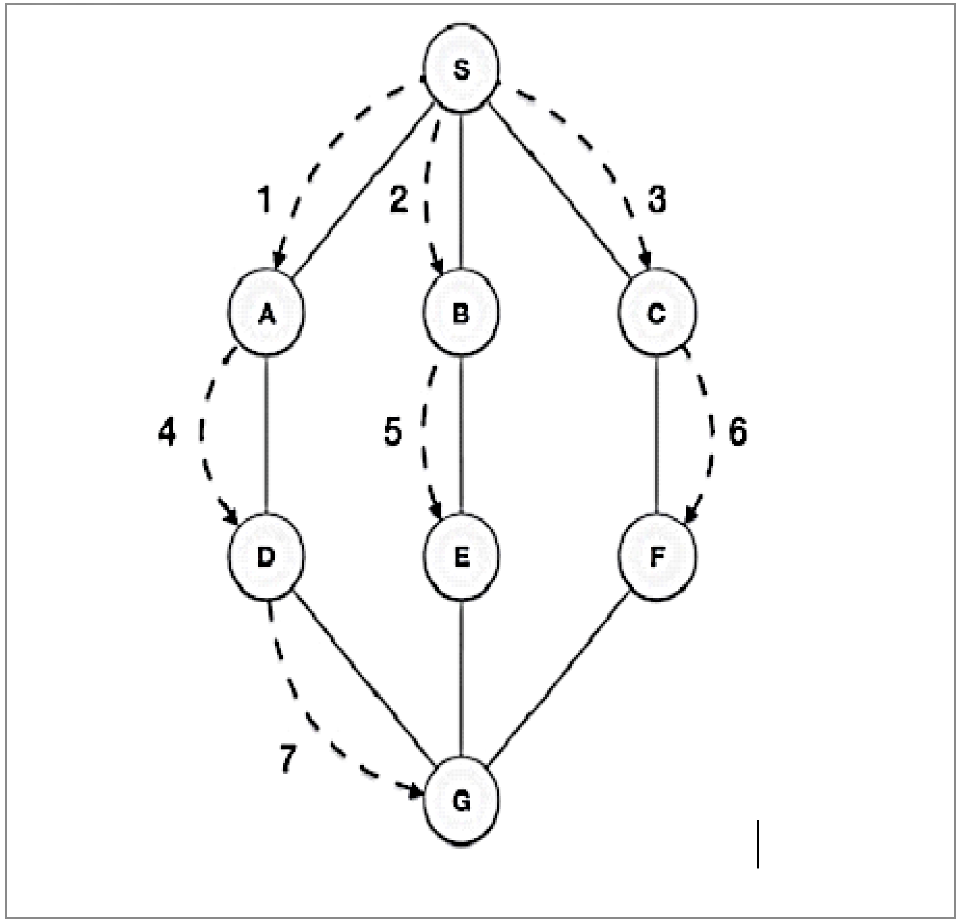
What is BFS?

(BFS) algorithm traverses a graph in a breadth-ward motion and uses a queue to remember to get the next vertex to start a search when a dead end occurs in any iteration.

BFS is basically a nodebased algorithm which is used to find the shortest path in the graph between two nodes. BFS moves through all of its nodes which are connected to the individual nodes.

BFS uses the FIFO (First In First Out) principle while using the Queue to find the shortest path. However, BFS is slower and requires a large memory space.

Example of BFS



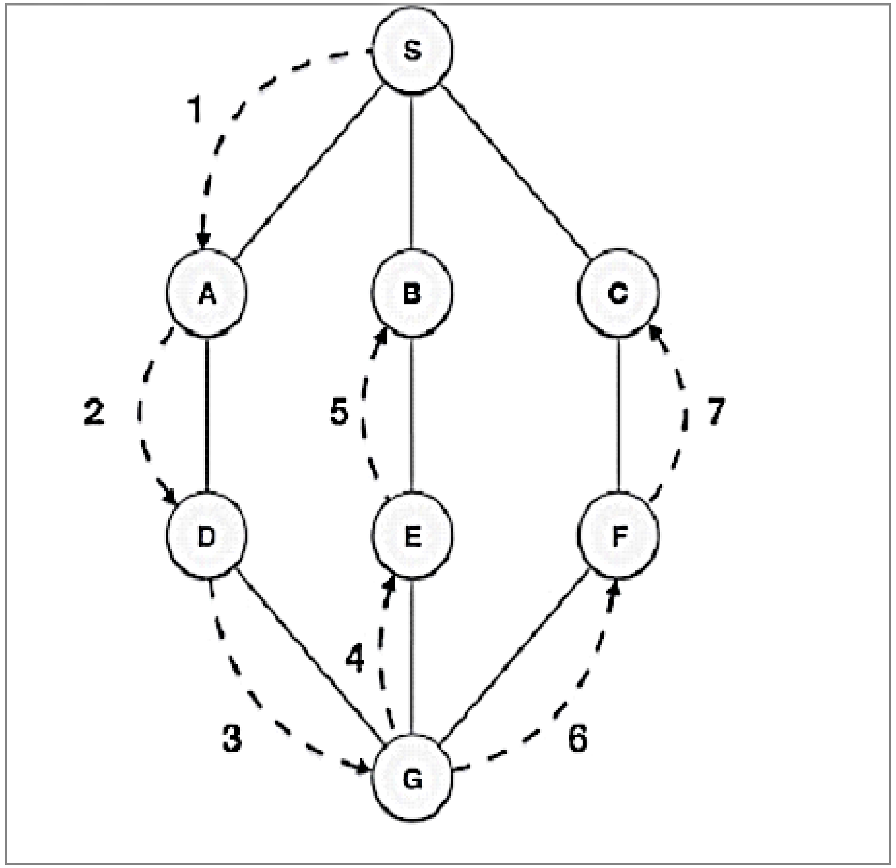
What is DFS?

Depth First Search (DFS) algorithm traverses a graph in a depth-ward motion and uses a stack to remember to get the next vertex to start a search when a deadend occurs in any iteration.

DFS uses LIFO (Last In First Out) principle while using Stack to find the shortest path. DFS is also called **Edge Based Traversal** because it explores the nodes along the edge or path. DFS is faster and requires less memory. DFS is best suited for decision trees.

.

Example of DFS



Difference between BFS and DFS

The following are the important differences between BFS and DFS –

Key	BFS	DFS
Definition	BFS stands for Breadth First Search.	DFS stands for Depth First Search.
Data structure	BFS uses a Queue to find the shortest path.	DFS uses a Stack to find the shortest path.
Source	BFS is better when target is closer to Source.	DFS is better when target is far from source.
Suitability for decision tree	As BFS considers all neighbor so it is not suitable for decision tree used in puzzle games.	DFS is more suitable for decision tree. As with one decision, we need to traverse further to augment the decision. If we reach the conclusion, we won.

Difference between BFS and DFS

The following are the important differences between BFS and DFS −

Key	BFS	DFS
Speed	BFS is slower than DFS.	DFS is faster than BFS.
Time Complexity	Time Complexity of BFS = O(V+E) where V is vertices and E is edges.	Time Complexity of DFS is also O(V+E) where V is vertices and E is edges.
Memory	BFS requires more memory space.	DFS requires less memory space.
Tapping in loops	In BFS, there is no problem of trapping into finite loops.	In DFS, we may be trapped into infinite loops.
Principle	BFS is implemented using FIFO (First In First Out) principle.	DFS is implemented using LIFO (Last In First Out) principle.

Conclusion

Both BFS and DFS are graph traversal algorithms. The most significant difference between the two is that the BFS algorithm uses a Queue to find the shortest path, while the DFS algorithm uses a Stack to find the shortest path.

Links and references

https://www.tutorialspoint.com/difference-between-bfs-and-dfs#:~:text=BFS%20is%20better%20when%20target,target%20is%20far%20from%20source.&text=As%20BFS%20considers%20all%20neighbor,more%20suitable%20for%20decision%20tree.