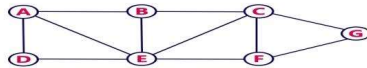
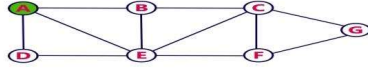


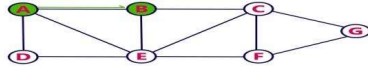
Consider the following example graph to perform DFS traversal



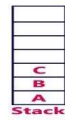
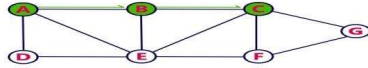
- Step 1:**
- Select the vertex **A** as starting point (visit **A**).
  - Push **A** on to the Stack.



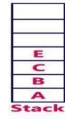
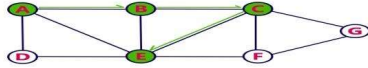
- Step 2:**
- Visit any adjacent vertex of **A** which is not visited (**B**).
  - Push newly visited vertex **B** on to the Stack.



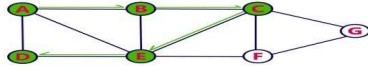
- Step 3:**
- Visit any adjacent vertex of **B** which is not visited (**C**).
  - Push **C** on to the Stack.



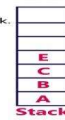
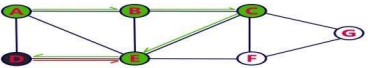
- Step 4:**
- Visit any adjacent vertex of **C** which is not visited (**E**).
  - Push **E** on to the Stack.



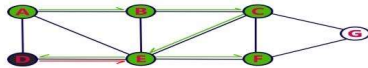
- Step 5:**
- Visit any adjacent vertex of **E** which is not visited (**D**).
  - Push **D** on to the Stack.



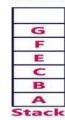
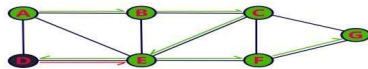
- Step 6:**
- There is no new vertex to be visited from **D**. So use back track.
  - Pop **D** from the Stack.



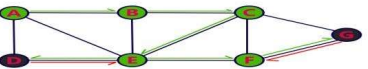
- Step 7:**
- Visit any adjacent vertex of **E** which is not visited (**F**).
  - Push **F** on to the Stack.



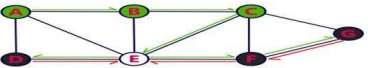
- Step 8:**
- Visit any adjacent vertex of **F** which is not visited (**G**).
  - Push **G** on to the Stack.



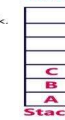
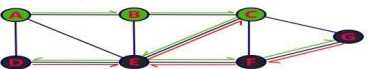
- Step 9:**
- There is no new vertex to be visited from **G**. So use back track.
  - Pop **G** from the Stack.



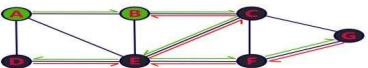
- Step 10:**
- There is no new vertex to be visited from **F**. So use back track.
  - Pop **F** from the Stack.



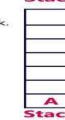
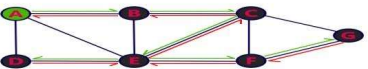
- Step 11:**
- There is no new vertex to be visited from **E**. So use back track.
  - Pop **E** from the Stack.



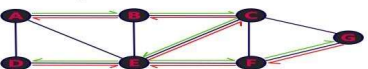
- Step 12:**
- There is no new vertex to be visited from **C**. So use back track.
  - Pop **C** from the Stack.



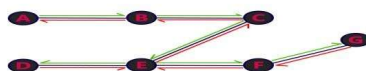
- Step 13:**
- There is no new vertex to be visited from **B**. So use back track.
  - Pop **B** from the Stack.



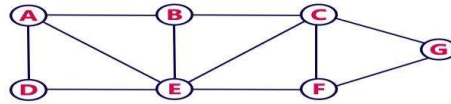
- Step 14:**
- There is no new vertex to be visited from **A**. So use back track.
  - Pop **A** from the Stack.



- Stack became Empty. So stop DFS Traversal.
- Final result of DFS traversal is following spanning tree.

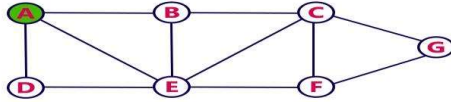


Consider the following example graph to perform BFS traversal



**Step 1:**

- Select the vertex **A** as starting point (visit **A**).
- Insert **A** into the Queue.

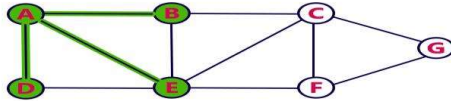


Queue



**Step 2:**

- Visit all adjacent vertices of **A** which are not visited (**D, E, B**).
- Insert newly visited vertices into the Queue and delete **A** from the Queue..

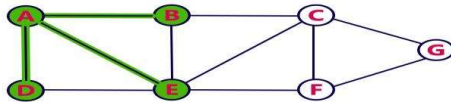


Queue



**Step 3:**

- Visit all adjacent vertices of **D** which are not visited (there is no vertex).
- Delete **D** from the Queue.

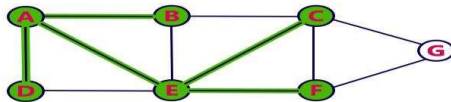


Queue



**Step 4:**

- Visit all adjacent vertices of **E** which are not visited (**C, F**).
- Insert newly visited vertices into the Queue and delete **E** from the Queue.

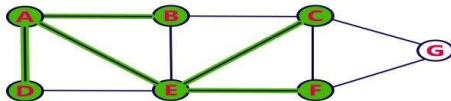


Queue



**Step 5:**

- Visit all adjacent vertices of **B** which are not visited (**there is no vertex**).
- Delete **B** from the Queue.

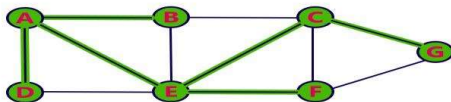


Queue



**Step 6:**

- Visit all adjacent vertices of **C** which are not visited (**G**).
- Insert newly visited vertex into the Queue and delete **C** from the Queue.

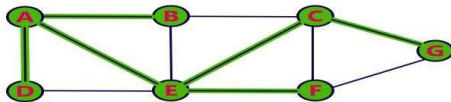


Queue



**Step 7:**

- Visit all adjacent vertices of **F** which are not visited (**there is no vertex**).
- Delete **F** from the Queue.

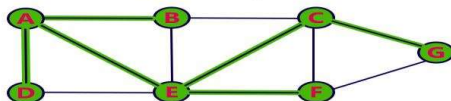


Queue



**Step 8:**

- Visit all adjacent vertices of **G** which are not visited (**there is no vertex**).
- Delete **G** from the Queue.



Queue



- Queue became Empty. So, stop the BFS process.
- Final result of BFS is a Spanning Tree as shown below...

