

1) What does DFS stand for?

- a) Directed First Search
- b) Delay First Search
- c) Depth Full Search
- d) Depth First Search

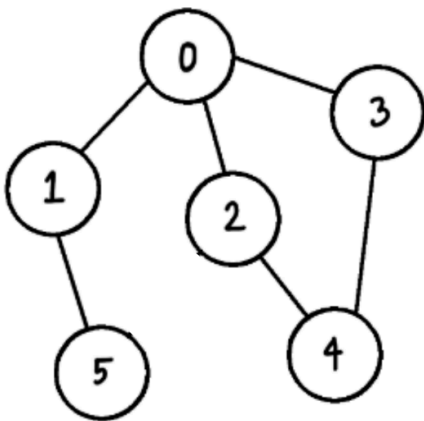
Explanation: DFS means Depth First Search.

2) How is DFS implemented?

- a. By Queue
- b. By Recursion
- c. Both A and B
- d. None of them

Explanation: DFS সাধারণত Recursion বা Stack ব্যবহার করে ইমপ্লিমেন্ট করা হয়। যদিও Stack ব্যবহার করে DFS করা যায়, তবে Recursion ব্যবহার করাই বেশি প্রচলিত। (Recursion ব্যবহার করে DFS আপনাদের মডিউলে দেখানো হয়েছে)

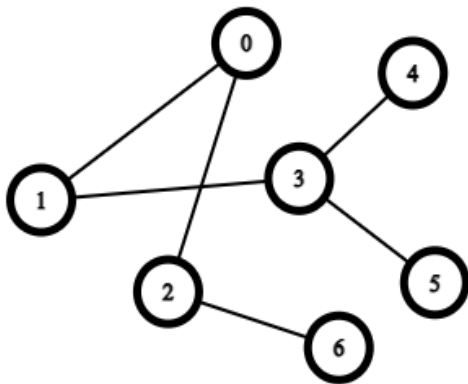
3) For the graph, When source=0, What following sequences of DFS will be?



- a. 0 -> 1-> 5 -> 2-> 4 -> 3
- b. 0 -> 1 -> 5 -> 3-> 4-> 2
- c. 0 -> 2 -> 4-> 3 -> 1-> 5
- d. All of a, b, and c.

Explanation: কুইজের a,b,c এই ৩ টা অপশনই সঠিক। (কিভাবে DFS ট্রাভার্স করতে হয় তা মডিউলে বিস্তারিত আলোচনা করা হয়েছে)

4) What will be the DFS traversal of the graph below if the source is 0?



a) 0 -> 3 -> 4 -> 5 -> 6 -> 2 -> 1

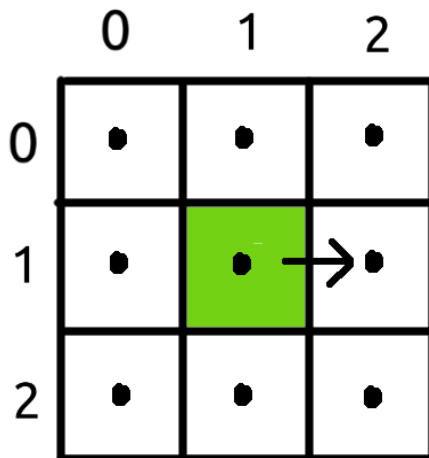
b) 0 -> 2 -> 6 -> 1 -> 3 -> 4 -> 5

c) 0 -> 1 -> 2 -> 3 -> 4 -> 5 -> 6

d) 0 -> 1 -> 3 -> 2 -> 6 -> 4 -> 5

Explanation: কুইজের b নং অপশন সঠিক। (কিভাবে DFS ট্রাভার্স করতে হয় তা মডিউলে বিস্তারিত আলোচনা করা হয়েছে)

5) In this 2D grid, we want to move from our source cell to the right. Our source cell $i=1$ and $j=1$. How can we do this?



a) By adding -1 to i and 0 to j

b) By adding 0 to i and -1 to j

c) By adding 0 to i and 1 to j

d) By adding 1 to i and 1 to j

Explanation: 2D গ্রিডে (i, j) কোঅর্ডিনেট ব্যবহার করে **move** করা হয়, যেখানে i = সারি (**row**) সংখ্যা এবং j = কলাম (**column**) সংখ্যা। ডানে যেতে হলে **row** পরিবর্তন হবে না তাই i এর সাথে 0 যোগ হবে এবং **column** এক ধাপ বাড়বে তাই j এর সাথে 1 যোগ হবে। তাই, $(i, j) \rightarrow (i + 0, j + 1)$ আমাদের ডানে যাওয়ার সঠিক উপায়।

6) In this 2D grid, we want to move from our source cell to the **bottom left** adjacent. Our source cell $i=1$ and $j=2$. How can we do this?

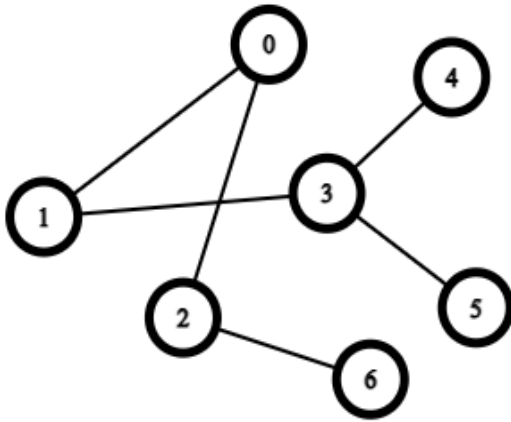
	0	1	2	3
0	•	•	•	•
1	•	•	•	•
2	•	•	•	•
3	•	•	•	•

- a) By adding -1 to i and -1 to j
- b) By adding 1 to i and -1 to j
- c) By adding -1 to i and 1 to j
- d) By adding 1 to i and 1 to j

Explanation: 2D গ্রিডে (i, j) কোঅর্ডিনেট ব্যবহার করে চলাচল করা হয়, যেখানে i = সারি (**row**) সংখ্যা এবং j = কলাম (**column**) সংখ্যা। **bottom-left** যেতে হলে **row** এর সাথে 1 যোগ করলে নিচে নামবে এবং **left** যেতে **column** এক ধাপ কমবে তাই j এর সাথে 1 বিয়োগ হবে।

তাই, আমাদের **bottom-left** যাওয়ার সঠিক উপায় $(i, j) \rightarrow (i + 1, j - 1)$

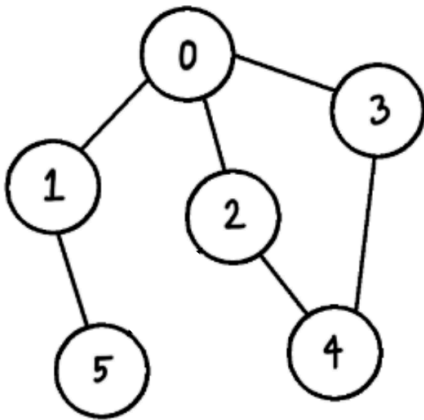
7) What will be the DFS traversal of the graph below if the source is 3?



- a) 3 -> 4 -> 5 -> 6 -> 2 -> 1 -> 0
- b) 3 -> 4 -> 5 -> 1 -> 0 -> 2 -> 6
- c) 3 -> 1 -> 2 -> 0 -> 4 -> 5 -> 6
- d) 3 -> 1 -> 0 -> 2 -> 4 -> 5 -> 6

Explanation: কুইজের b নং অপশন সঠিক। (কিভাবে DFS ট্রাভার্স করতে হয় তা মডিউলে বিস্তারিত আলোচনা করা হয়েছে)

8) For the graph When source=5; What following sequences of DFS will be?



- A. 5 -> 1 -> 0 -> 2 -> 4 -> 3
- B. 5 -> 1 -> 0 -> 3 -> 4 -> 2
- C. Both A and B
- D. None of them

Explanation: কুইজের A,B এই 2 টা অপশনই সঠিক। (কিভাবে DFS ট্রাভার্স করতে হয় তা মডিউলে বিস্তারিত আলোচনা করা হয়েছে)

9) What type of traversal method is there for the DFS graph?

- A) Level-wise Traversal
- B) Breadth-wise Traversal
- C) Depth-wise Traversal
- D) Random Traversal

Explanation: DFS এ সাধারণত Depth-wise traversal পদ্ধতি অনুসরণ করে, যেখানে প্রথমে একটি নোড select করা হয়, তারপর যতদূর সম্ভব Depth এ যায়।

10) When does DFS backtracking occur?

- A) When all adjacent nodes are visited
- B) When a new node is found
- C) When a new edge is discovered
- D) When there is only one node

Explanation: DFS-এ যখন কোনো নোডের সকল Adjacent নোড ভিসিট করা শেষ হয় এবং নতুন কোনো নোডে যাওয়ার path থাকে না। তখন এটি পূর্বের নোডে ফিরে যায় বা backtracking ঘটে।

ভুল Option গুলোর ব্যাখ্যা:

✗ B) When a new node is found

→ নতুন নোড পাওয়া গেলে ব্যাকট্র্যাকিং হয় না, বরং DFS সেই নোডে যায় এবং নোড ভিসিট করে। backtracking তখনই ঘটে যখন কোনো নতুন নোড পাওয়া যায় না।

✗ C) When a new edge is discovered

→ নতুন কোনো Edge discover হলে DFS সেই edge ধরে নতুন নোডে যায়, ব্যাকট্র্যাকিং তখন হয় না। ব্যাকট্র্যাক তখনই হয় যখন আর কোনো নতুন নোডে যাওয়ার edge or path থাকে না।

✗ D) When there is only one node

→ যদি গ্রাফে শুধু একটি নোড থাকে, তাহলে ব্যাকট্র্যাক করার প্রয়োজনই নেই কারণ নতুন কোনো নোডে যাওয়ার সুযোগ নেই। DFS শুধুমাত্র একটি নোড ভিজিট করেই শেষ হয়ে যাবে।