

Directed Graphs

[Back to Week 1](#)


3/3 points earned (100%)

Quiz passed!

1 / 1
points

1.

```

1 (seed = 489125)
2 Consider the adjacency-lists representation of a digraph with 8 vertices and
3 13 edges:
4 A: E
5 B: A G
6 C: B
7 D: C G
8 E: F
9 F: A B
10 G: H F C
11 H: D
12
13
14 Here is a graphical representation of the same digraph:
15
16 (A)----->(B)----->(C)----->(D)
17   ^         ^         ^         ^
18   |         |         |         |
19   |         |         |         |
20   |         |         |         |
21   |         |         |         |
22   |         |         |         |
23   |         |         |         |
24   |         |         |         |
25   |         |         |         |
26   |         |         |         |
27   |         |         |         |
28   |         |         |         |
29   |         |         |         |
30   |         |         |         |
31   v         v         v         v
32 (E)----->(F)----->(G)----->(H)
33

```

Run breadth-first search (using the adjacency-lists representation), starting from vertex A. Give the sequence in which the vertices are dequeued from the FIFO queue.

Your answer should be a sequence of uppercase letters, starting with A.

Correct Response

```
1 The correct answer is: A E F B G H C D
2
3 Here is a trace of the breadth-first search:
4
5 enqueue A
6 dequeue A
7   enqueue E
8 dequeue E
9   enqueue F
10 dequeue F
11   check A
12   enqueue B
13 dequeue B
14   check A
15   enqueue G
16 dequeue G
17   enqueue H
18   check F
19   enqueue C
20 dequeue H
21   enqueue D
22 dequeue C
23   check B
24 dequeue D
25   check C
26   check G
27
28
29 Here are the shortest paths and distances:
30
31 A to A (0): A
32 A to B (3): A->E->F->B
33 A to C (5): A->E->F->B->G->C
34 A to D (6): A->E->F->B->G->H->D
35 A to E (1): A->E
36 A to F (2): A->E->F
37 A to G (4): A->E->F->B->G
38 A to H (5): A->E->F->B->G->H
```

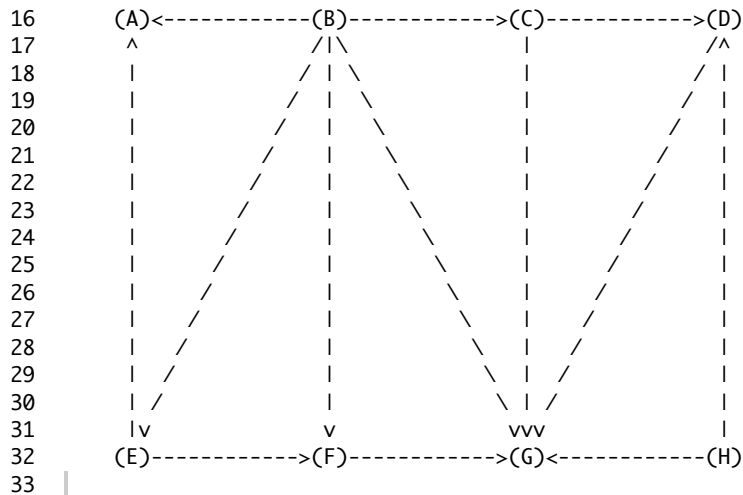


1 / 1
points

2.

```
1 (seed = 979780)
2 Consider the adjacency-lists representation of a DAG with 8 vertices and 13
  edges:
3
4   A:
5   B: C A F E G
6   C: D G
7   D: G
8   E: A F
9   F: G
10  G:
11  H: G D
```

14 Here is a graphical representation of the same DAG:



Give the topological order of the vertices that results from the DFS-based topological sort algorithm. As usual, perform the first DFS from vertex A.

Your answer should be a sequence of 8 uppercase letters.

H B E F C D G A



Correct Response

```
1 The correct answer is: H B E F C D G A
2
3 Here is a trace of the depth-first search:
4
5 dfs(A)
6 A done
7 dfs(B)
8   dfs(C)
9     dfs(D)
10       dfs(G)
11         G done
12       D done
13     check G
14   C done
15 check A
16 dfs(F)
17   check G
18 F done
19 dfs(E)
20   check A
21   check F
22 E done
23 check G
24 B done
25 check C
26 check D
27 check E
28 check F
29 check G
30 dfs(H)
31   check G
32   check D
33 H done
34
35
36 The postorder is the order in which the vertices are done. The reverse
37 postorder
38 provides a topological order.
```



1 / 1
points

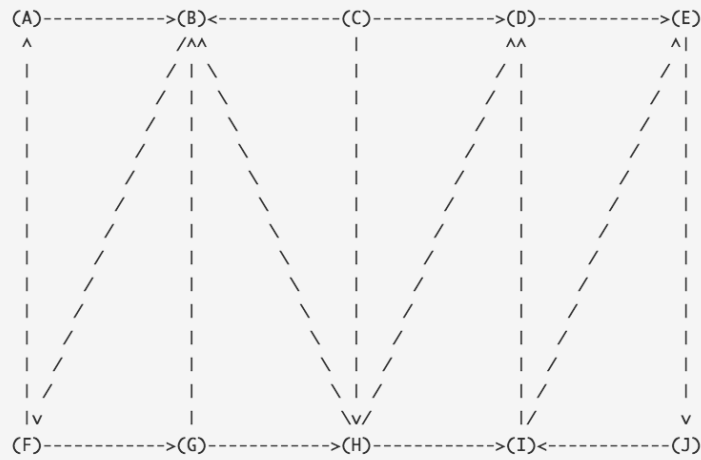
3.

(seed = 584559)

Consider the adjacency-lists representation of a digraph G with 10 vertices and 17 edges:

```
A: B
B: F
C: H D B
D: E
E: J
F: A G
G: B H
H: I B D
I: E D
J: I
```

Here is a graphical representation of the same digraph G :



Compute the strongly-connected components of the digraph using the Kosaraju-Sharir algorithm. Assume that the first depth-first search of Kosaraju-Sharir computes the reverse postorder of G^R :

D I J E A F B H C G

Give the sequence of the 10 integers in the `id[]` array for the vertices A through J.

```

v    A B C D E F G H I J
-----
id[v]
```

1 1 2 0 0 1 1 1 0 0



Correct Response

```

1 The correct answer is: 1 1 2 0 0 1 1 1 0 0
2
3
4   v   A B C D E F G H I J
5 -----
6 id[v] 1 1 2 0 0 1 1 1 0 0
7
8 The second depth-first search considers the vertices in the following
  order:
9
10    D I J E A F B H C G
11
12 Here is a trace of the second depth-first search:
13
14
15 strong component 0
16 -----
17 dfs(D)
18   dfs(E)
19     dfs(J)
20       dfs(I)
21         check E
22         check D
23       I done
24     J done
25   E done
26 D done
27 -----
28
29 check I
30 check J
31 check E
32
33 strong component 1
34 -----
35 dfs(A)
36   dfs(B)
37     dfs(F)
38       check A
39     dfs(G)
40       check B
41     dfs(H)
42       check I
43       check B
44       check D
45     H done
46   G done
47 F done
48 B done
49 A done
50 -----
51
52 check F
53 check B
54 check H
55
56 strong component 2
57 -----
58 dfs(C)
59   check H
60   check D
61   check B
62 C done
63 -----
64
65 check G

```

