CE DFS vs BFS CODE   MISSING 10 Possible Points

| 11/21/2022

Attempt 1 ⌄	**IN PROGRESS** Next Up: Submit Assignment	🗨 Add Comment

---

Unlimited Attempts Allowed

⌄ **Details**

<div align="center">

Graphs

CE: DFS vs BFS CODE

✔

Learning Objectives

</div>

- Familiarize yourself with class BreadthFirstPaths.
- Compare the paths that are found using depth first search and breadth first search.

<div align="center">

🔭

Overview

</div>

In this CE, you are provided an image of a simple graph. Based on that graph, you will create paths using both the depth first search (dfs) and breadth first search (bfs) algorithms. The path will be listed side by side for easier comparison.

**Getting Started:**

- Create a package called **graphDFSvsBFS**
- Create a graph file **SimpleGraph.txt** based on the graph shown below.
  Use the format described in tinyG.txt (**https://algs4.cs.princeton.edu/41graph/** ↪
  (https://algs4.cs.princeton.edu/41graph/))
  List the edges in ascending order. Begin with all the edges that start with 0, then the
  edges that start with 1, etc. Sort the edges that start with the same vertex by the second
  vertex.
  *Why does it matter? Listing the edges in a systematic order reduces the chance of
  overlooking an edge or adding it twice.*
- Create a class **DFSvsBFS** that includes the main method.
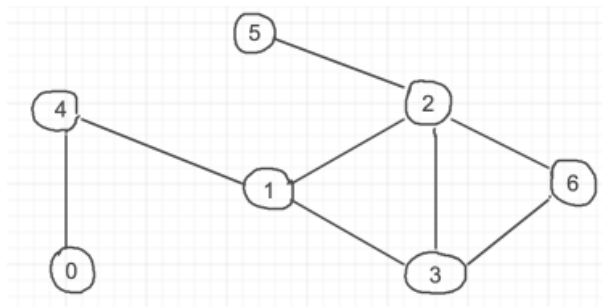
In the main method do the following:

- Create a graph based on the graph file SimpleGraph.txt.
- Print the adjacency list.
  It should be formatted as shown in the output.
- Create a variable called source, and assign vertex 1 as the source.
- Find two paths from the source to each of the vertices.
  First using depth first search (dfs), then using breadth first search (bfs).
  The path should be listed side by side in straight columns as shown in the expected
  output.
  *Important: As always, resist the temptation to use hard-coded paths. When I change the
  graph file the adjacency list and the paths need to change accordingly.*
- Compare and contrast the paths found by dfs and bfs.

< Previous

**(https://slcc.instructure.com
/courses/817632/modules/items
/18753078)**

Submit Assignment

Next >

**(https://slcc.instructure.com
/courses/817632/modules/items
/18753081)**

Expected Output

```
Adjacency List:
---------------
0: 4
1: 4->3->2
2: 6->5->3->1
3: 6->2->1
4: 1->0
5: 2
6: 3->2


Paths DFS:              Shortest Paths BFS:
---------------         -------------------
1..4..0                 1..4..0
1                       1
1..3..6..2              1..2
1..3                    1..3
1..4                    1..4
1..3..6..2..5           1..2..5
1..3..6                 1..3..6
```

Submission

Create a screen recording following the **guidelines for lab recordings (https://slcc.instructure.com/courses/817632/pages/guidelines-for-ce-recordings)** .
The video should be **25 - 50 seconds** long.

🚀 (https://slcc.instructure.com/courses/817632/modules/items/18753071)

🚀 (https://slcc.instructure.com/courses/817632/modules/items/18753072)

🚀 (https://slcc.instructure.com/courses/817632/modules/items/18753073)

🚀 (https://slcc.instructure.com/courses/817632/modules/items/18753074)

🚀 (https://slcc.instructure.com/courses/817632/modules/items/18753075)

🔗 (https://slcc.instructure.com/api/v1/courses/817632/module_item_redirect/18753077)

🔗 (https://slcc.instructure.com/api/v1/courses/817632/module_item_redirect/18753078)

📍📝 (https://slcc.instructure.com/courses/817632/modules/items/18753080)

📝 (https://slcc.instructure.com/courses/817632/modules/items/18753081)

Choose a submission type

T Text

⬆ Upload

Office 365

⋮ More

Previous (https://slcc.instructure.com /courses/817632/modules/items /18753078)

Submit Assignment

Next ❯ (https://slcc.instructure.com /courses/817632/modules/items /18753081)