CSIT 503 HW7
Topic: Breadth-First Search

Dr. Boxiang Dong

1 Problem Description

Instructions. You are provided one skeleton program named *Graph.java*. The source files are available on Canvas in a folder named *HW7*. Please modify the skeleton code to solve the following tasks.

- Task 1 (100 pts). Implement the *bfs(int s)* function as discussed in Lecture 15.
  - **Note:** You **should** return an integer array that records the minimum distance between every node to the source s.
  - **Hint 1:** The colors have been defined in the class for you to use.
  - **Hint 2:** In the matrix-adjacency representation, each node is just an integer. So you cannot do this
    u.color = WHITE
    for a node u. Instead, I suggest you to create an integer array to represent the colors of the nodes, another array to represent the distance d for the nodes.
  - **Hint 3:** You can ignore the parent π in your code.
  - **Hint 4:** To use the Enqueue and Dequeue function, you may use your previous implementation of Queue in HW3. Or you can use the add() and *remove()* function of Java LinkedList (`https://docs.oracle.com/javase/7/docs/api/java/util/LinkedList.html`). More tutorials can be found `https://www.javatpoint.com/java-linkedlist` and `http://www.javadb.com/using-a-queue-or-linkedlist/`.
- Task 2 (Extra Credit 100 pts). Implement the *DFS()* and *DFS_Visit()* functions as discussed in Lecture 16.

2 Submission Guideline

1. Work individually.
2. Please directly insert your code in the appropriate place in Graph.java.

3. Create a zip file of your .java source programs and submit it on Canvas on time. A late penalty of 10 points for each late day applies. Any late for more than three days receives zero automatically.