Data Structures and Algorithms
Assignment 4
(100 points)

Instructions
Submit your code in Canvas. Include a brief README file explaining what you did. Please put down your name in the comments at the beginning of all the source code files.

Program Requirements (100 points)
In this assignment you will implement two basic graph algorithms: **breadth-first search** and **depth-first traversal**. We are not providing any code template, but you must follow the input and output formats specified below.

First, implement a graph data structure using **adjacency matrix**. Your code must read graphs of arbitrary size from a file (as long as there is enough memory available, your code must work: do not assume a maximum number of vertices). The input format is as follows: one line per vertex, **the first element is the vertex id and the following numbers are the adjacent vertices**. The input file (e.g., a txt file) will always be a bunch of integers (1, 2, 3, etc) and white space. For example,

```
1 2 3 4
2 1 5
3 1 4
4 1 3
5 2
```

is a graph with five vertices, and five undirected edges: (1, 3), (1, 2), (1, 4), (2, 5) and (3, 4).

Then, implement two algorithms (each one is worth 50 points):
- Breadth-first search (BFS). Assume the starting node is 1, print all the vertices using BFS.
- Depth-first traversal (DFS). Assume the starting node is 1, print all the vertices using BFS.

A few comments:
- Look carefully at the sample input and output files and follow the same format.
- When you look through vertices, visit them in increasing order.

NOTE: You need to generate your own test cases to ensure your programs successfully run and give the correct outputs.