Lab01

UNINFORMED SEARCH

1. Description

You are given a graph and a pair of **source** – **destination** nodes taken from the graph. Write a small program to find a possible path from the **source** node to the **destination** node using **Breath-first search (BFS)** and **Depth-first search (DFS)** algorithms.

You are asked to run BFS and DFS on at least FIVE different graphs of more than 10 nodes to make a comprehensive comparison of these algorithms' performance regarding the following aspects:

- The number of nodes visited
- The length of the discovered paths
- The space complexity.

Hint:

- *You should set some limits to prevent DFS from traversing too many nodes when there is no path or the path is too complex to be found. For example, stop DFS when it reaches a maximum depth or after a period of time.*
- *Graphs demonstrating different types of path (i.e. the destination is near or far from the source) are useful for your analysis.*

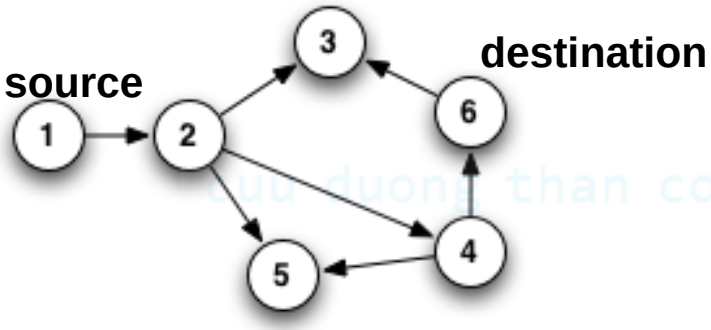
2. Specifications

- **Input:** the given graph is represented by its adjacency matrix, which is stored in the file **input.txt**. The input file format is described as follows:
  - The first line contains an integer N, which is the number of nodes in the graph.
  - The second line stores a triple of integers describing the **source** node index, the **destination** node index (indices start from 0) and the chosen algorithm (0 = BFS, 1 = DFS). Numbers are separated by white spaces.
  - N next lines represent the N × N adjacency matrix. Each line contains N integers. The number at [i, j] (row i, column j) determines whether there is a link

from node i to node j, [i, j] = 1 if there is such a link and otherwise [i, j] = 0). Numbers are separated by white spaces.

- **Output**: the result is stored in the file **output.txt**, whose format is described as follows:
  - If a path from the source to the destination exists:
    - The first line contains an integer C, which is the cost of the discovered path (i.e. the number of steps).
    - The second line show in order all the nodes on the path using their indices. Numbers are separated by white spaces.
  - Otherwise: the first line contains an optional notification.
- The **main function** must perform the following basic actions
  - Read the input data from the input file and store it in appropriate data structures.
  - Call the function **BFS** (or **DFS**), which implements the Breath-first search (or Depth-first search) algorithm, to find a path from **source** to **destination**.
  - Show the outputs.
- When there are many candidate nodes with equal possibilities, **the algorithms must visit them following their ascending index ordering**.
- Use relative paths for the input and output file and do not change their names. Your program needs these requirements to be graded automatically.

An example of the input graph and its corresponding file **input.txt**.

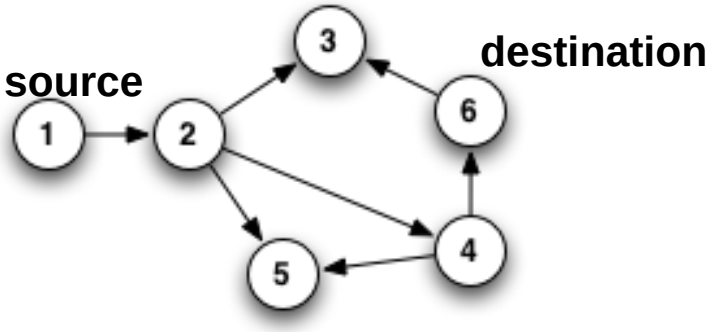



```
6
0 5 0
0 1 0 0 0 0
0 0 1 1 1 0
0 0 0 0 0 0
0 0 0 0 1 1
0 0 0 0 0 0
0 0 1 0 0 0
```

The program will create the below **output.txt** file after running **BFS.**

```
3
0 1 3 5
```

Another example with the same input graph yet a different pair of source – destination.




```
6
5 0 1
0 1 0 0 0 0
0 0 1 1 1 0
0 0 0 0 0 0
0 0 0 0 1 1
0 0 0 0 0 0
0 0 1 0 0 0
```

The program will create the below **output.txt** file after running **DFS.**

```
No path is found.
```

3. Grading

No.	Specifications	Scores
1	Read the input data and successfully store it in some data structures	20%
2	Implement most of the BFS algorithm	10%
3	Implement most of the DFS algorithm	10%
4	Correctly discover the path using BFS	10%
5	Correctly discover the path using DFS	10%
6	Follow the lab specifications well	10%
7	A comprehensive comparison of BFS and DFS	30%
Total		100%

4. Notice

- This is an **INDIVIDUAL** assignment.
- 10% bonus will be given as an award for students who can submit a perfect solution within 5 days.
- You are allowed to use data structure functions/libraries (e.g. queue, stack), yet **you must implement the uninformed search algorithms by yourself**.
- Report can be written in English or Vietnamese.