Fall 2023 – CS 303 Algorithms and Data Structures
Lab 08

Notes:

• Implement the algorithm and analyze the results using the give input files

• Deliverables: Report.pdf file and your code file (please do not send a zip file. If you have more than one class in your code, then submit each file separately through Canvas.)

• Homework report must follow the guidelines provided in the sample report uploaded in Canvas

Objectives:

- Implement an undirected graph using adjacency lists
- Implement Breadth First Search algorithm on graphs.

Problems

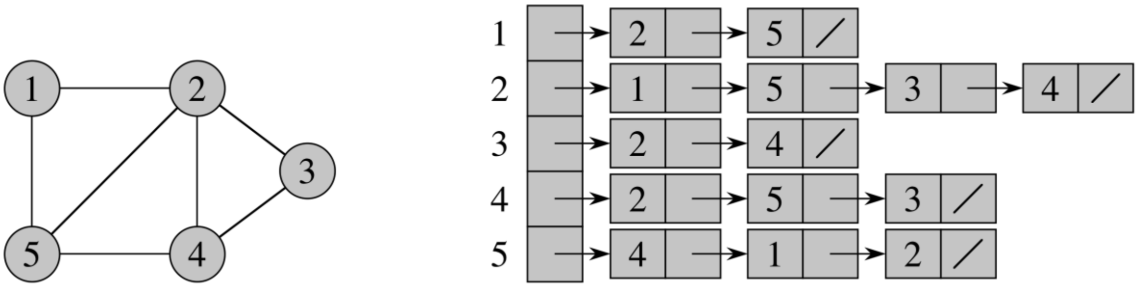
1. Write a program to implement an undirected graph using adjacency lists. Write a driver program to read the input files (mediumG.txt and largeG.txt) that contain the number of vertices, number of edges, and the corresponding list of edges and create an undirected graph. You can verify that the graph was constructed successfully by displaying the graph in the form of adjacency list for each vertex (see example below).

2. Implement BFS algorithm on the undirected graphs (created in Problem 1) following the pseudo-code given in the next page. Print the BFS paths from a source (you can pick any vertex as the source) to all the other nodes in the graph.

3. Write a detailed report on the performance of the BFS algorithm for the two input files.

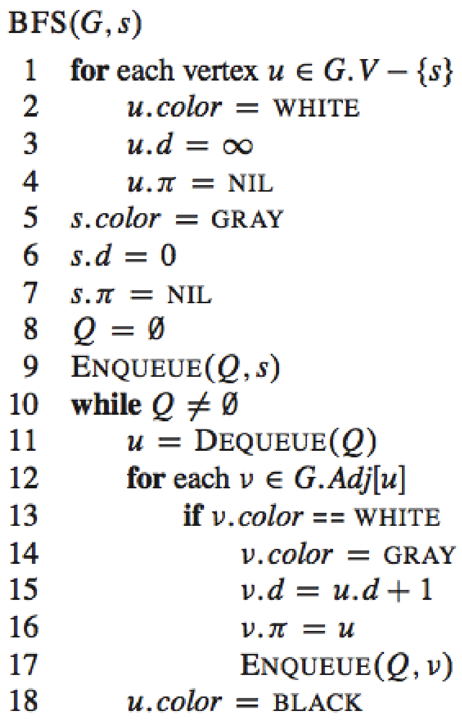
Note that the BFS algorithm uses the queue data structure. You can use any existing package (such as linked list or queue) in the language of your choice to implement the queue, there is no need to implement the queue from scratch.

Adjacency List Visualization

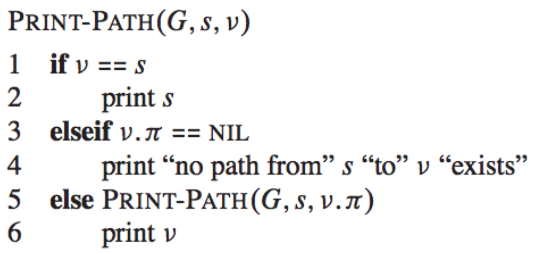


Output:

```
1: 2, 5

2: 1, 5, 3, 4

3: 2, 4

4: 2, 5, 3

5: 4, 1, 2
```



```
BFS(G, s)
for each vertex $u \in G.V - \{s\}$
    $u.color$ = WHITE
    $u.d = \infty$
    $u.\pi$ = NIL
$s.color$ = GRAY
$s.d = 0$
$s.\pi$ = NIL
$Q = \emptyset$
ENQUEUE(Q, s)
while $Q \neq \emptyset$
    $u$ = DEQUEUE(Q)
    for each $v \in G.Adj[u]$
        if $v.color$ == WHITE
            $v.color$ = GRAY
            $v.d = u.d + 1$
            $v.\pi = u$
            ENQUEUE(Q, v)
    $u.color$ = BLACK
```



```
PRINT-PATH(G, s, v)
if $v == s$
    print $s$
elseif $v.\pi$ == NIL
    print "no path from" $s$ "to" $v$ "exists"
else PRINT-PATH(G, s, $v.\pi$)
    print $v$
```

DATA

largeG.txt

medium.txt

Grading Rubric

Coding	Implementing Algorithms	20 points
	Producing Correct Outputs	20 points
Report	Explaining the algorithms used	10 points
	Displaying the output with a graph or table	20 points
	Comparing the outputs and discussing the time complexity of algorithms	20 points
	Correct submissions of the files (ICF, Code.zip, report.pdf)	10 points