

**Gebze Technical University
Computer Engineering**

CSE 222 - 2019 Spring

HOMEWORK 8 REPORT

**OZAN ŞELTE
161044061**

Course Assistant: Ayşe Şerbetçi Turan

1 INTRODUCTION

1.1 Problem Definition

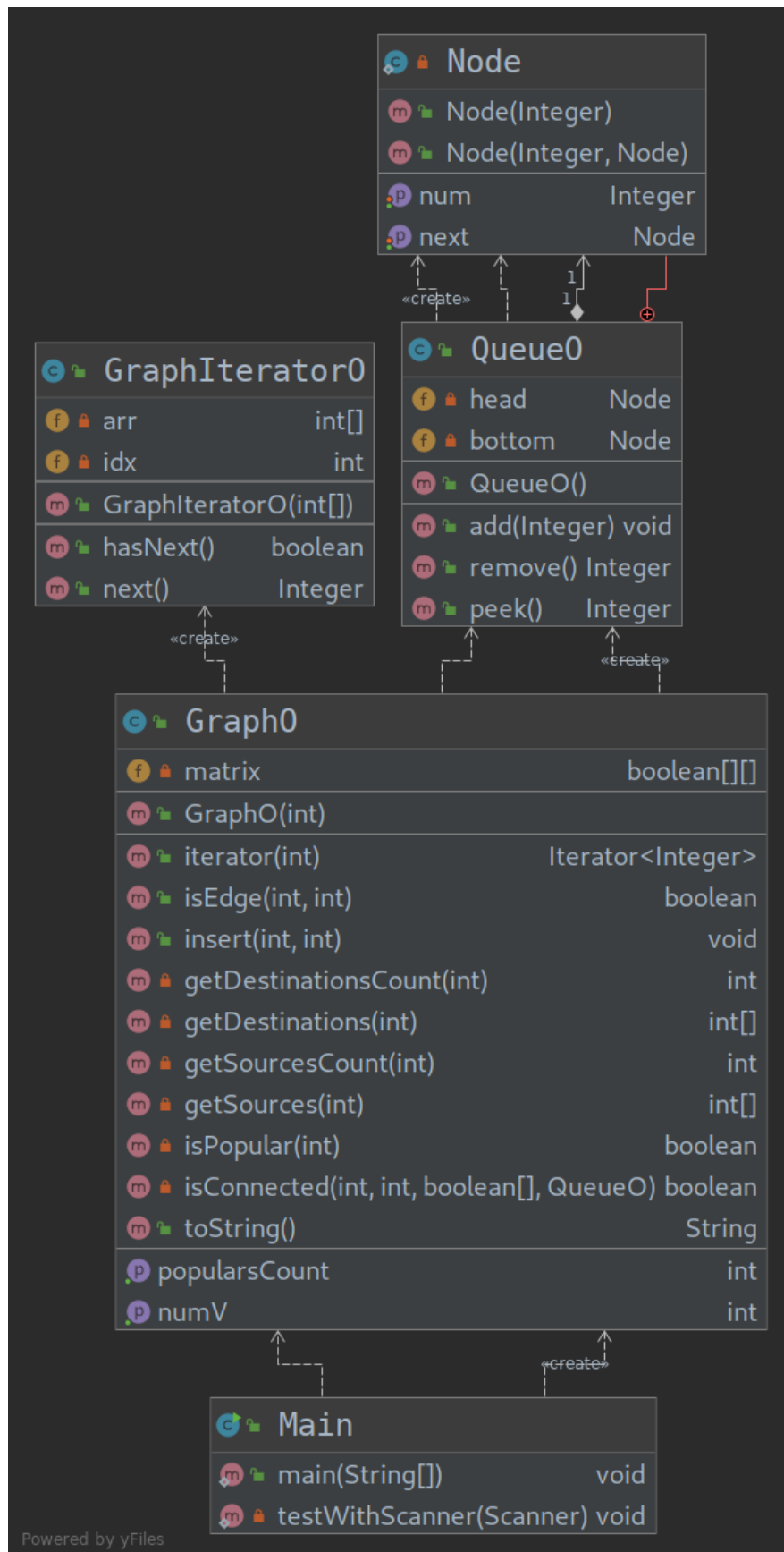
The problem is to count the number of fully-pointed vertices in a given graph. We have to read an input file and create a graph. After that, we need to count the pointed ones.

1.2 System Requirements

A computer, JDK and a terminal.

2 METHOD

2.1 Class Diagrams



2.2 Use Case Diagrams

Not required.

2.3 Problem Solution Approach

I have used adjacency matrix to implement Graph and BFS(breadth first search) to count the popular ones. We are not allowed to use any class in the Java's Collections package so I have implemented Queue data structure too for BFS.

Node: All of the methods including constructors have $O(1)$.

QueueO: All of the methods including constructors have $O(1)$.

GraphIteratorO: next and hasNext methods have $O(1)$ but constructor has $O(n)$ because it is created a new array according to the given array.

GraphO class:

toString(): It has $O(n)$ because it creates a String with 2D matrix.

getDestinationsCount(): It has $O(n)$. It returns the number of edges from given vertice.

getDestinations(): It has $O(n)$. It returns the array of the sources of the edges from given vertice.

getSourcesCount(): It has $O(n)$. It returns the number of edges to given vertice.

getSources(): It has $O(n)$. It returns the array of the destinations of the edges to given vertice.

isConnected(): It has $O(n)$. It is the BFS method. It returns true if there is a directed path from the given vertice to another given one, else otherwise. It check a vertices only one time and runs recursively.

isPopular(): It has $O(n^2)$. It calls isConnected for every vertice to the given one.

GraphO(): The constructor has $O(n^2)$, it is preparing an $N \times N$ matrix.

getNumV(): It has $O(1)$. Returns a data field.

iterator(): It has $O(n^2)$. It calls GraphIteratorO's constructor and returns it.

isEdge(): It has $O(1)$. It just return the matrix' given field.

getPopularsCount(): It calls the isPopular method for every vertice. It has $O(n^3)$.

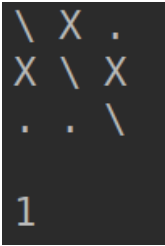
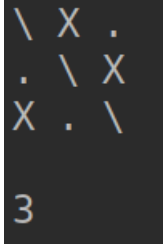
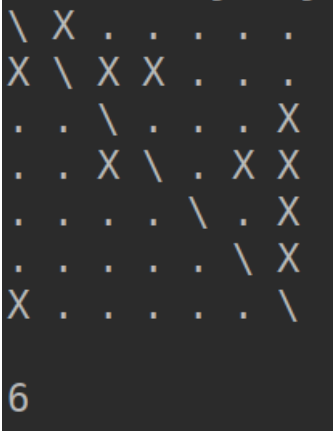
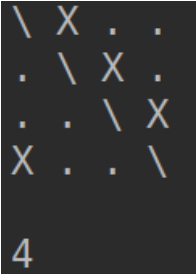
insert(): It has $O(1)$. It just sets the matrix' given field as True.

3 RESULT

3.1 Test Cases

I have implemented four custom input files to see what happens at critical situations.

3.2 Running Results

3 3 1 2 2 1 2 3	
3 3 1 2 2 3 3 1	
7 11 1 2 2 1 2 3 4 6 4 3 2 4 3 7 4 7 5 7 6 7 7 1	
4 4 1 2 2 3 3 4 4 1	
4 3 1 3 2 4 3 2	