Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна

Факультет комп’ютерних наук

Кафедра моделювання систем і технологій

**ЛАБОРАТОРНА РОБОТА №1**

З дисципліни: «Об’єктно-орієнтоване програмування»

Тема: Трикутник Флойда

Виконав студент 2 курсу

групи КС-21

Клочко А.В.

Перевірив:

к.т.н., доц. Нагорний К.А.

Харків – 2020

# РОЗДІЛ 1 ВСТУП

**Тема:** Розробка додатку для побудови трикутника Флойда (Floyd's triangle).

**Мета роботи:** Набути навики застосування базових принципів об’єктно-орієнтованої розробки програмних систем (ПС), а саме: абстракція (abstraction) та інкапсуляція (encapsulation). Набути базові навики проектування статичної структури класів ПС із використанням мови UML (class diagram).

**Завдання:** на основі параметру, кількість строк, побудувати трикутник Флойда в прямому та інверсному (зворотному) порядку. Вивести побудовані трикутники на консоль. Побудувати діаграму класів в нотації UML.

# РОЗДІЛ 2 РЕЗУЛЬТАТИ РОБОТИ

## 2.1 UML – діаграма класів

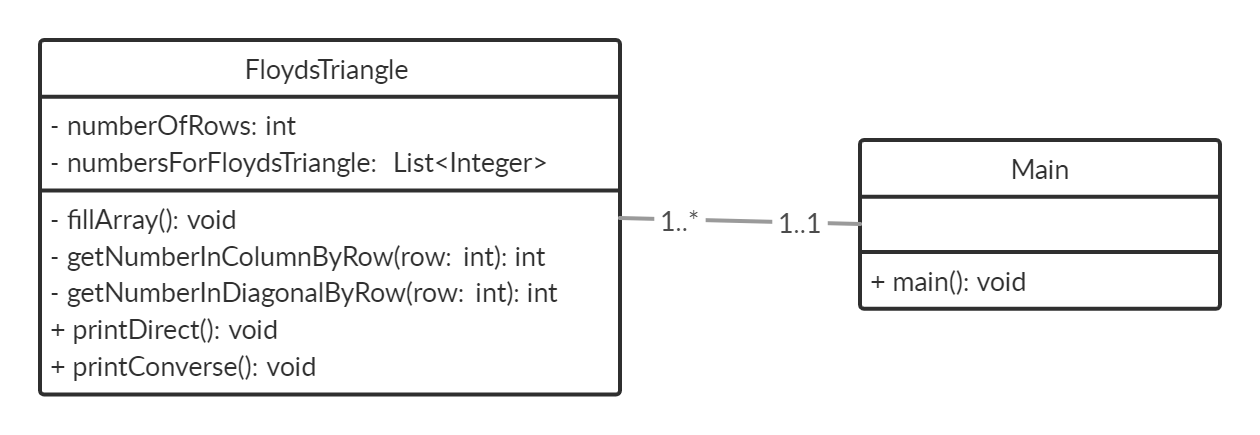


Рисунок 2.1 – UML – діаграма класів Трикутника Флойда

## 2.2 Лістинг вихідного коду

Лістинг класу «Main»

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

FloydsTriangle floydsTriangle;

String choice;

while(true){

System.out.println("1: Make Floyd's Triangle\n" +

"0: Exit");

choice = scan.next();

if(choice.equals("1")){

System.out.println("1: Print direct triangle\n" +

"2: Print converse triangle");

choice = scan.next();

System.out.print("Enter the number of rows for floyd's triangle: ");

int numberOfRowsFloydsTriangle = scan.nextInt();

floydsTriangle = new FloydsTriangle(numberOfRowsFloydsTriangle);

if(choice.equals("1")){

floydsTriangle.printDirect();

}else if(choice.equals("2")) {

floydsTriangle.printConverse();

}

}else if(choice.equals("0")){

break;

}

}

}

}

Лістинг класу «FloydsTriangle»

import java.util.ArrayList;

import java.util.List;

public class FloydsTriangle {

private final int numberOfRows;

private List<Integer> numbersForFloydsTriangle;

public FloydsTriangle(int numberOfRows){

this.numberOfRows = numberOfRows;

fillArray();

}

private void fillArray(){

numbersForFloydsTriangle = new ArrayList<Integer>(numberOfRows);

for (int i = 0; i < numberOfRows; i++) {

for (int j = getNumberInColumnByRow(i); j < getNumberInDiagonalByRow(i+1)+1; j++) {

numbersForFloydsTriangle.add(j);

}

}

}

private static int getNumberInColumnByRow(int row){

return (row\*row + row + 2) / 2;

}

private static int getNumberInDiagonalByRow(int row){

return (row\*(row + 1)) / 2;

}

public void printDirect(){

if(numberOfRows != 0){

int index = 0;

for (int i = 0; i < numberOfRows; i++) {

for (int j = 0; j < i+1; j++) {

System.out.print(numbersForFloydsTriangle.get(index)+" ");

index++;

}

System.out.println();

}

}

}

public void printConverse(){

if(numberOfRows != 0){

int index = numbersForFloydsTriangle.size()-1;

for (int i = numberOfRows; i > 0; i--) {

for (int j = i; j > 0 ; j--) {

System.out.print(numbersForFloydsTriangle.get(index)+" ");

index--;

}

System.out.println();

}

}

}

}

## 2.3 Результати розрахунків

Прямо надрукуємо трикутник Флойда для п’яти строк:

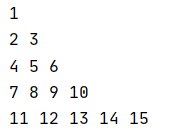


Рисунок 2.3.1 – Результати прямого друку при кількості строк 5

Обернено надрукуємо трикутник Флойда для п’яти строк:

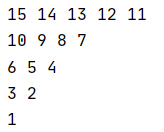


Рисунок 2.3.2 – Результати оберненого друку при кількості строк 5