Cod sursa

public class Conversion {

private static int[][] adjacencyMatrix;

private static int[] predecessorList;

private static int[] list;

private static int nodes;

private static void printMatrix() {

int i, j;

for (i = 1; i <= nodes; i++) {

for (j = 1; j <= nodes; j++) {

System.out.print(adjacencyMatrix[i][j] + " ");

}

System.out.println();

}

}

private static void printList(int[] list, int length) {

for (int i = 1; i <= length; i++) {

System.out.print(list[i] + " ");

}

System.out.println();

}

private static void init() throws IOException {

Scanner in = new Scanner(System.in);

System.out.println("Numarul de varfuri");

nodes = in.nextInt();

adjacencyMatrix = new int[nodes + 1][nodes + 1];

System.out.println("Introduceti matricea de adiacenta");

for (int i = 1; i <= nodes; i++) {

for (int j = 1; j <= nodes; j++) {

adjacencyMatrix[i][j] = in.nextInt();

}

}

predecessorList = new int[nodes + 2];

list = new int[nodes \* (nodes - 1) + 2];

in.close();

}

public static void main(String[] arg) {

int i, j;

try {

init();

} catch (IOException e) {

System.out.println("Eroare la initializare");

e.printStackTrace();

}

int nr = 1;

for (j = 1; j <= nodes; j++) {

predecessorList[j] = nr;

for (i = 1; i <= nodes; i++)

if (adjacencyMatrix[i][j] != 0) {

list[nr] = i;

nr++;

}

}

System.out.println("Matricea de adiacenta");

printMatrix();

System.out.println("Sirul predecesorilor");

printList(predecessorList, nodes);

System.out.println("Sirul pozitilor predecesorilor");

printList(list, nr - 1);

}

}