

**Ahsanullah University of Science and Technology (AUST)**

**Department of Computer Science and Engineering**

**Course No. : CSE2202**

**Course Title : Numerical Methods Lab**

**Date of Performance: 10.02.2019**

**Date of Submission: 3.03.2019**

**Name :Drubojit Saha**

**ID : 17.01.04.027**

**Section : A2**

Online 4:Implementation of GaussElimination Method

package gausseliminationtest;

import java.util.Scanner;

public class GaussEliminationTest {

public static double coeffecient[][] = new double[10][10];

public static double rightsidevector[] = new double[10];

public static double resultvector[] = new double[10];

public static int flag;

public static void gauss(int size) {

double pivot, factor, sum;

for (int k = 1; k <= size - 1; k++) {

pivot = coeffecient[k][k];

if (pivot < 0.000001) {

flag = 0;

break;

}

flag= 1;

for (int i = k + 1; i <= size; i++) {

factor = coeffecient[i][k] / pivot;

for (int j = k + 1; j <= size; j++) {

coeffecient[i][j] = coeffecient[i][j] - factor \* coeffecient[k][j];

}

rightsidevector[i] = rightsidevector[i] - factor \* rightsidevectorvector[k];

}

}

resultvector[size] = rightsidevector[size] / coeffecient[size][size];

for (int k = size - 1; k >= 1; k--) {

sum = 0.0;

for (int j = k + 1; j <= size; j++) {

sum = sum + coeffecient[k][j] \* resultvector[j];

}

resultvector[k] = (rightsidevector[k] - sum) / coeffecient[k][k];

}

}

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

int size;

System.out.println("Enter the size of the system: ");

size = input.nextInt();

System.out.println("one row at a time");

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

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

coeffecient[i][j] = input.nextDouble();

}

}

System.out.println("Enter the input vectors: ");

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

rightsidevectorvector[i] = input.nextDouble();

}

gauss(size);

if (flag != 0) {

System.out.println("Solution vector: ");

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

System.out.println(resultvector[i]);

}

} else {

System.out.println("No solution vector");

}

}

}