**נספח**

**התוכנית הבאה מריצה את שיטת ג'קובי על מטריצה נתונה מתוך קובץ**

// jacobi1.java - jacobi iterative method

import java.util.Scanner;

public class jacobi1

{

static int test\_convergence(double []x, double []oldx, int n)

{

double maxvalue, tempvalue;

int i;

double epsilon = 0.000000001;

maxvalue = 0.0;

for(i=0; i < n; i++)

{

tempvalue = Math.abs(x[i] - oldx[i]);

if (tempvalue > maxvalue)

maxvalue = tempvalue;

} // for

if (maxvalue > epsilon)

return 0;

else

return 1;

} // test\_convergence

static void jacobi(double A[][], double b[], int n, double x[])

{

int i, j, k, p;

double []oldx;

oldx = new double [n];

do {

for (i=0; i < n; i++)

oldx[i] = x[i];

for (i=0; i < n; i++)

{

x[i] = b[i];

for(j=0; j < n; j++)

if ( i != j)

x[i] = x[i] - A[i][j] \* oldx[j];

x[i] = x[i]/A[i][i];

} // for

} while(test\_convergence(x, oldx,n) == 0);

} // jacobi

static void print\_result(double A[][],

double x[], double b[], int n)

{

int i, j;

double sum;

System.out.print("Solution X:\n");

for(i=0; i < n; i++)

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

System.out.println();

for(i=0; i < n; i++)

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

System.out.println("\n Verification:");

for(i=0; i < n; i++)

{

sum = A[i][0] \* x[0];

System.out.print(A[i][0]+ " \* " + x[0] + " ");

for(j=1; j < n; j++)

{

sum = sum + A[i][j] \* x[j];

System.out.print(" + " + A[i][j] + " \* " +x[j] + " ");

} // for

System.out.println();

System.out.println(" = " + sum + " ?= " + b[i]);

} // for

} // print\_result

static void read\_file(Scanner fp, double [][]A, double []b, int n)

{

int i, j;

for(i=0; i < n; i++)

for(j=0; j < n; j++)

A[i][j] = fp.nextDouble();

for(i=0; i < n; i++)

b[i] = fp.nextDouble();

} // read\_file

static void print\_original\_system(double [][]A, double []b, int n)

{

int i, j;

String temp;

System.out.println("Original System:");

for(i=0; i < n; i++)

{

temp = "";

for(j=0; j < n; j++)

temp += " " + A[i][j] + " ";

System.out.println(temp + " " + b[i] + " ");

} // for

} // print\_original\_system

public static void main(String args[])

{

int i, n;

double A[][], b[], x[];

Scanner sc;

if (args.length < 1)

{

System.out.println("Usage: gaussian filename\n");

return;

} // if

try {

sc = new Scanner(new java.io.File(args[0]));

}

catch(java.io.FileNotFoundException e)

{

System.out.println("File not found");

return;

}// catch

n = sc.nextInt();

A = new double [n+1][n];

b = new double [n];

x = new double [n];

read\_file(sc, A, b, n);

jacobi(A, b, n, x);

System.out.println(" A x = b");

print\_original\_system(A, b, n);

print\_result(A, x, b, n);

} // main

} // jacobi1

אם נריץ את התוכנית על הקובץ mat8.txt שתוכנו:

4

4 1 -1 1

2 7 3 1

1 -2 8 2

1 0 1 -4

7

6

-1

-6

פלט ריצה:

A x = b

Original System:

4.0 1.0 -1.0 1.0 7.0

2.0 7.0 3.0 1.0 6.0

1.0 -2.0 8.0 2.0 -1.0

1.0 0.0 1.0 -4.0 -6.0

Solution X:

X[0] X[1] X[2] X[3]

1.072072071854296 0.5450450454298463 -0.5315315316207099 1.6351351349054735

Verification:

4.0 \* 1.072072071854296 + 1.0 \* 0.5450450454298463 + -1.0 \* -0.5315315316207099 + 1.0 \* 1.6351351349054735

= 6.999999999373214 ?= 7.0

2.0 \* 1.072072071854296 + 7.0 \* 0.5450450454298463 + 3.0 \* -0.5315315316207099 + 1.0 \* 1.6351351349054735

= 6.00000000176086 ?= 6.0

1.0 \* 1.072072071854296 + -2.0 \* 0.5450450454298463 + 8.0 \* -0.5315315316207099 + 2.0 \* 1.6351351349054735

= -1.000000002160129 ?= -1.0

1.0 \* 1.072072071854296 + 0.0 \* 0.5450450454298463 + 1.0 \* -0.5315315316207099 + -4.0 \* 1.6351351349054735

**התוכנית הבאה מריצה את שיטת גאוס-זיידל על מטריצה נתונה בקובץ כפרמטר לתוכנית ראשית**

// gs1.java - gaus\_seidel iterative method

import java.util.Scanner;

public class gs1

{

static void gaus\_seidel(double A[][], double b[], int n, double x[])

{

int i, j, k, p;

double []oldx;

oldx = new double [n];

do {

for (i=0; i < n; i++)

oldx[i] = x[i];

for (i=0; i < n; i++)

{

x[i] = b[i];

for(j=0; j < n; j++)

if ( i != j)

x[i] = x[i] - A[i][j] \* x[j];

x[i] = x[i]/A[i][i];

} // for

} while(test\_convergence(x, oldx,n) == 0);

} // gaus\_seidel

static int test\_convergence(double []x, double []oldx, int n)

{

double maxvalue, tempvalue;

int i;

double epsilon = 0.000000001;

maxvalue = 0.0;

for(i=0; i < n; i++)

{

tempvalue = Math.abs(x[i] - oldx[i]);

if (tempvalue > maxvalue)

maxvalue = tempvalue;

} // for

if (maxvalue > epsilon)

return 0;

else

return 1;

} // test\_convergence

static void print\_result(double A[][],

double x[], double b[], int n)

{

int i, j;

double sum;

System.out.print("Solution X:\n");

for(i=0; i < n; i++)

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

System.out.println();

for(i=0; i < n; i++)

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

System.out.println("\n Verification:");

for(i=0; i < n; i++)

{

sum = A[i][0] \* x[0];

System.out.print(A[i][0]+ " \* " + x[0] + " ");

for(j=1; j < n; j++)

{

sum = sum + A[i][j] \* x[j];

System.out.print(" + " + A[i][j] + " \* " +x[j] + " ");

} // for

System.out.println();

System.out.println(" = " + sum + " ?= " + b[i]);

} // for

} // print\_result

static void read\_file(Scanner fp, double [][]A, double []b, int n)

{

int i, j;

for(i=0; i < n; i++)

for(j=0; j < n; j++)

A[i][j] = fp.nextDouble();

for(i=0; i < n; i++)

b[i] = fp.nextDouble();

} // read\_file

static void print\_original\_system(double [][]A, double []b, int n)

{

int i, j;

String temp;

System.out.println("Original System:");

for(i=0; i < n; i++)

{

temp = "";

for(j=0; j < n; j++)

temp += " " + A[i][j] + " ";

System.out.println(temp + " " + b[i] + " ");

} // for

} // print\_original\_system

public static void main(String args[])

{

int i, n;

double A[][], b[], x[];

Scanner sc;

if (args.length < 1)

{

System.out.println("Usage: gaussian filename\n");

return;

} // if

try {

sc = new Scanner(new java.io.File(args[0]));

}

catch(java.io.FileNotFoundException e)

{

System.out.println("File not found");

return;

}// catch

n = sc.nextInt();

A = new double [n+1][n];

b = new double [n];

x = new double [n];

read\_file(sc, A, b, n);

gaus\_seidel(A, b, n, x);

System.out.println(" A x = b");

print\_original\_system(A, b, n);

print\_result(A, x, b, n);

} // main

} // gs1

אם נריץ את התוכנית על הקובץ mat8.txt שתוכנו:

4

4 1 -1 1

2 7 3 1

1 -2 8 2

1 0 1 -4

7

6

-1

-6

**פלט ריצה:**

A x = b

Original System:

4.0 1.0 -1.0 1.0 7.0

2.0 7.0 3.0 1.0 6.0

1.0 -2.0 8.0 2.0 -1.0

1.0 0.0 1.0 -4.0 -6.0

Solution X:

X[0] X[1] X[2] X[3]

1.0720720719853558 0.5450450449560018 -0.531531531580318 1.6351351351012595

Verification:

4.0 \* 1.0720720719853558 + 1.0 \* 0.5450450449560018 + -1.0 \* -0.531531531580318 + 1.0 \* 1.6351351351012595

= 6.999999999579002 ?= 7.0

2.0 \* 1.0720720719853558 + 7.0 \* 0.5450450449560018 + 3.0 \* -0.531531531580318 + 1.0 \* 1.6351351351012595

= 5.99999999902303 ?= 6.0

1.0 \* 1.0720720719853558 + -2.0 \* 0.5450450449560018 + 8.0 \* -0.531531531580318 + 2.0 \* 1.6351351351012595

= -1.000000000366673 ?= -1.0

1.0 \* 1.0720720719853558 + 0.0 \* 0.5450450449560018 + 1.0 \* -0.531531531580318 + -4.0 \* 1.6351351351012595

= -6.0 ?= -6.0