**נספח**

התוכנית הבאה מחשבת את האינטגרל של f(x) = x2 בקטע (0,2) בשיטת הטרפז. ב-1000000 נקודות קיבלנו 9 ספרות משמעותיות.

// trapezoid1.java - implement trapezoid integration

class Function

{

static double f(double x)

{

return x\*x;

} //

static double If( double x)

{

return x\*x\*x/3.0;

} // If

}// Function

public class trapezoid1

{

static double trapezoid( double a, double b, int n)

{

double h, I, t;

int i;

h = (b - a)/n;

I = (Function.f(a)+ Function.f(b))/2.0;

t = a;

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

{

t = t + h;

I = I + Function.f(t);

} /\* for \*/

I = I\*h;

return I;

} // trapezoid

public static void main(String args[])

{

System.out.println("Integral x\*\*2(0,2)");

System.out.println("real value: " +

(Function.If(2.0)- Function.If(0.0)) +

", Trapezoid: " + trapezoid(0, 2.0,100000));

} // main

} //trapezoid1

פלט ריצה:

Integral x\*\*2(0,2)

real value: 2.6666666666666665, Trapezoid: 2.666666666794974

התוכנית הבאה מחשבת את האינטגרל של f(x) = x2 בקטע (0,2) בשיטת המלבן. ב-1000000 נקודות קיבלנו 9 ספרות משמעותיות.

// midpoint1.java - implement midpoint integration

class Function

{

static double f(double x)

{

return x\*x;

} //

static double If( double x)

{

return x\*x\*x/3.0;

} // If

}// Function

public class midpoint1

{

static double midpoint( double a, double b, int n)

{

double h, I, t;

int i;

h = (b - a)/n;

I = 0;

t = a+h/2.0;

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

{

I = I + Function.f(t);

t = t + h;

} // for

I = I\*h;

return I;

} // midpoint

public static void main(String args[])

{

System.out.println("Integral x\*\*2(0,2)");

System.out.println("real value: " +

(Function.If(2.0)- Function.If(0.0)) +

", Midpoint: " + midpoint(0, 2.0,100000));

} // main

} //midpoint1

פלט ריצה:

Integral x\*\*2(0,2)

real value: 2.6666666666666665, Midpoint: 2.6666666665949847

התוכנית הבאה מפעילה את סכמת הטרפז:

// trapezoid2.java - implement trapezoid integration

class Function

{

static double f(double x)

{

return x\*x;

} //

static double If( double x)

{

return x\*x\*x/3.0;

} // If

}// Function

public class trapezoid2

{

static double trapezoid( double a, double b, int n)

{

double h, I, t;

int i;

h = (b - a)/n;

I = (Function.f(a)+ Function.f(b))/2.0;

t = a;

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

{

t = t + h;

I = I + Function.f(t);

} /\* for \*/

I = I\*h;

return I;

} // trapezoid

static double trapez\_schema(double a, double b, double eps)

{

int n;

double t1, t2;

n = 128;

t2 = trapezoid(a, b, n);

do {

n = n \* 2;

t1 = t2;

t2 = trapezoid(a, b, n);

} while(Math.abs(t1 - t2) > eps);

return t2;

} // trapez\_schema

public static void main(String args[])

{

System.out.println("Integral x\*\*2(0,2)");

System.out.println("real value: " +

(Function.If(2.0)- Function.If(0.0)) +

", Trapezoid\_Schema: " + trapez\_schema(0, 2.0,0.000000001));

} // main

} //trapezoid2

פלט ריצה:

Integral x\*\*2(0,2)

real value: 2.6666666666666665, Trapezoid\_Schema: 2.6666666669771075

התוכנית הבאה מממשת את סכמת השילוב:

// combi1.java - implement combi integration

class Function

{

static double f(double x)

{

return x\*x;

} //

static double If( double x)

{

return x\*x\*x/3.0;

} // If

}// Function

public class combi1

{

static double combi( double a, double b, double eps)

{

double h, t, T,M;

int i, k, n;

h = b - a;

n = 1;

T = h\*(Function.f(a) + Function.f(b))/2.0;

for(k=0; k < 30; k++)

{

M=0;

t = a+0.5\*h;

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

{

M = M + Function.f(t);

t = t + h;

} // for

M = M\*h;

T = (T+M)/2.0;

if ( Math.abs(T-M) < eps )

return T;

h = h/2.0;

n = n\*2;

} // for

return T;

} // combi

public static void main(String args[])

{

System.out.println("Integral x\*\*2(0,2)");

System.out.println("real value: " +

(Function.If(2.0)- Function.If(0.0)) +

", Combi: " + combi(0, 2.0,0.00000001));

} // main

} //combi1

פלט ריצה:

Integral x\*\*2(0,2)

real value: 2.6666666666666665, Combi: 2.66666666790843

התוכנית הבאה מבצעת אותה אינטגרציה בשיטת סימפסון:

// simpson1.java - implement simpson integration

class Function

{

static double f(double x)

{

return x\*x;

} //

static double If( double x)

{

return x\*x\*x/3.0;

} // If

}// Function

public class simpson1

{

static double simpson( double a, double b, int n)

{

double h, I, t, fact;

int i;

h = (b - a)/n;

I = Function.f(a)+ Function.f(b);

t = a;

fact = 4.0;

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

{

t = t + h;

I = I + fact \* Function.f(t);

if (fact == 4.0)

fact = 2.0;

else

fact = 4.0;

} // for

I = I\*h/3;

return I;

} // simpson

public static void main(String args[])

{

System.out.println("Integral x\*\*2(0,2)");

System.out.println("real value: " +

(Function.If(2.0)- Function.If(0.0)) +

", Simpson: " + simpson(0, 2.0,100000));

} // main

} //simpson1

פלט ריצה:

Integral x\*\*2(0,2)

real value: 2.6666666666666665, Simpson: 2.6666666666616323