Задание на Лр №5. Численное интегрирование.

1. Вычислить интеграл с помощью метода прямоугольников.

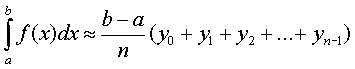
2. Вычислить интеграл с помощью метода трапеций.

3. Вычислить интеграл с помощью метода Симпсона.

4. Вычислить интеграл с помощью метода Монте-Карло.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Но\_мер варианта |  | a | b | Но\_мер вари\_анта |  | a | b |
| 1 |  | 1 | 6 | 2 |  | 1 | 7 |
| 3 |  | 1 | 7 | 4 |  | -3 | 3 |
| 5 |  | 1 | 5 | 6 |  | 0 | 3 |
| 7 |  | 0.5 | 2.4 | 8 |  | 0 | 3 |
| 9 |  | -0.5 | 4.5 | 10 |  | -2 | 2 |

**//Интегрирование методом прямоугольников.**



//Драйвер интегрирования методом прямоугольников

#include <stdio.h>

#include <stdlib.h>

#include<math.h>

float F(float x);

float f1(float x);

float Ipr(float x[],float y[],int n);

int main()

{

int i;

const int n=20;

float x[n],y[n],dy[n];

float r,s;

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

{

x[i]=0.1\*i;

y[i]=F(x[i]);

}

r=Ipr(x,y,n);

s=f1(2);

printf("I= %f II= %f\n",r,s);

system("pause");

return 0;

}

//Функция вычисления значений

float F(float x)

{return (- x\*x+2\*x);}

//Функция вычисления интеграла

float f1(float x)

{return(x\*x\*(1-x/3.));}

//Функция численного интегрирования методом прямоугольников

float Ipr(float x[],float y[],int n)

{ int i;

float s=0;

for(i=0; i<n-1; i++)

s+=y[i];

return (x[n-1]-x[0])\*s/n;

}

**//Интегрирование методом трапеций**

//Драйвер интегрирования методом трапеций

#include <stdio.h>

#include <stdlib.h>

#include<math.h>

float F(float x);

float f1(float x);

float Itr(float x[],float y[],int n);

int main()

{

int i;

const int n=20;

float x[n],y[n],dy[n];

float r,s;

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

{

x[i]=0.1\*i;

y[i]=F(x[i]);

printf(" i= %i, x= %f y=%f \n",i,x[i],y[i]);

}

r=Itr(x,y,n);

s=f1(2);

printf("I= %f II= %f\n",r,s);

system("pause");

return 0;

}

//Функция вычисления значений

float F(float x)

{return (- x\*x+2\*x);}

//Функция вычисления интеграла

float f1(float x)

{return(x\*x\*(1-x/3.));}

//Функция численного интегрирования методом трапеций

float Itr(float x[],float y[],int n)

{ int i;

float s=(y[0]+y[n])/2.;

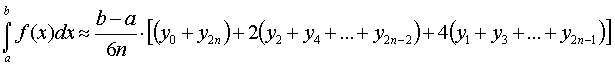
for(i=1; i<n; i++)

s+=y[i];

return (x[n]-x[0])\*s/n;

}

//**Интегрирование методом Симпсона**



//Драйвер интегрирование методом Симпсона

#include <stdio.h>

#include <stdlib.h>

#include<math.h>

float F(float x);

float f1(float x);

float Isim(float x[],float y[],int n);

int main()

{

int i;

const int n=20;

float x[n],y[n];

float r,s;

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

{

x[i]=0.1\*i;

y[i]=F(x[i]);

}

r=Isim(x,y,n);

s=f1(2);

printf("I= %f II= %f\n",r,s);

system("pause");

return 0;

}

//Функция вычисления значений

float F(float x)

{return (- x\*x+2\*x);}

//Функция вычисления интеграла

float f1(float x)

{return(x\*x\*(1-x/3.));}

//Функция численного интегрирования методом Симпсона

float Isim(float x[],float y[],int n)

{ int i;

float s=0,s1=0;

for(i=1; i<n-1; i+=2)

s+=y[i];

for(i=2; i<n-2; i+=2)

s1+=y[i];

s=(y[0]+y[n-1])+2.\*s+4.\*s1;

return (x[n-1]-x[0])\*s/n/6.;

}

//**Численное интегрирование методом Монте-Карло.**

S f(x)dx=(b-a)/N \*Sum(f(xj), где xj=a+yj(b-a)

yj --случайная величина равномерно

распределенная на интервале [0-1].

//Драйвер отладки и тестирования функции

// интегрирования методом Монте -Карло

#include <stdio.h>

#include <stdlib.h>

#include<math.h>

float F(float x);

float f1(float x);

float Ipr(float x[],float y[],int n);

float Imk(float a,float b,int n);

int main()

{

int i;

const int n=100000;

float x[n],y[n],dy[n];

float r,s,a,b;

a=0;b=2;

r=Imk(a,b,n);

printf("I= %f \n",r);

system("pause");

return 0;

}

//Функция вычисления значений

float f(float x)

{return (- x\*x+2\*x);}

//Функция численного интегрирования методом Монте-Карло.

float Imk(float a,float b,int n)

{ int i;

float s=0,r=b-a;

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

s+=f(a+(float)rand()/32767.\*r);

return (r\*s/n);

}