Московский государственный технический университет им. Н. Э. Баумана

(МГТУ им. Баумана)



*Факультет «Фундаментальные науки»*

*Кафедра «Математическое моделирование»*

**Лабораторный практикум**

**по дисциплине «Информатика»**

**Лабораторная работа №6**

***Вариант №17***

Выполнил

Проверила

Проверила

*Студент группы*

*ФН12-11Б*

*к.т.н., доцент*

*кафедры ИУ-7*

*к.т.н., доцент кафедры ИУ-7*

*Моисеенко Б. В.*

*Дебривная Т. Л.*

*Серебрякова И. Л.*

**Условие задачи**

Вычислить значение определенного интеграла от выбранной подынтегральной функции и заданных с клавиатуры интервалов интегрирования, точности и двух заданных значений начальных разбиений 4-мя методами.

1. Метод центральных прямоугольников:
2. Метод трапеций:
3. Метод кубических парабол:

Условие: n – кратно 3.

1. Метод Уэддля:

Условие: n – кратно 6.

**Описание данных**

*Входные данные:*

a, b – интервал интегрирования, тип double (по условию)

eps – точность вычислений, тип double (по условию)

split1, split2 – начальные разбиения, тип longint (в случае больших чисел)

*Промежуточные вычисления:*

h – шаг разбиения при интегрировании, тип extended (вещественное число)

x – переменная, принимающая значения от a до b при интегрировании, тип extended (вещественное число)

i - счетчик циклов, longint (в случае больших чисел)

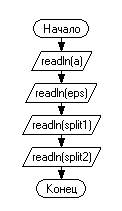
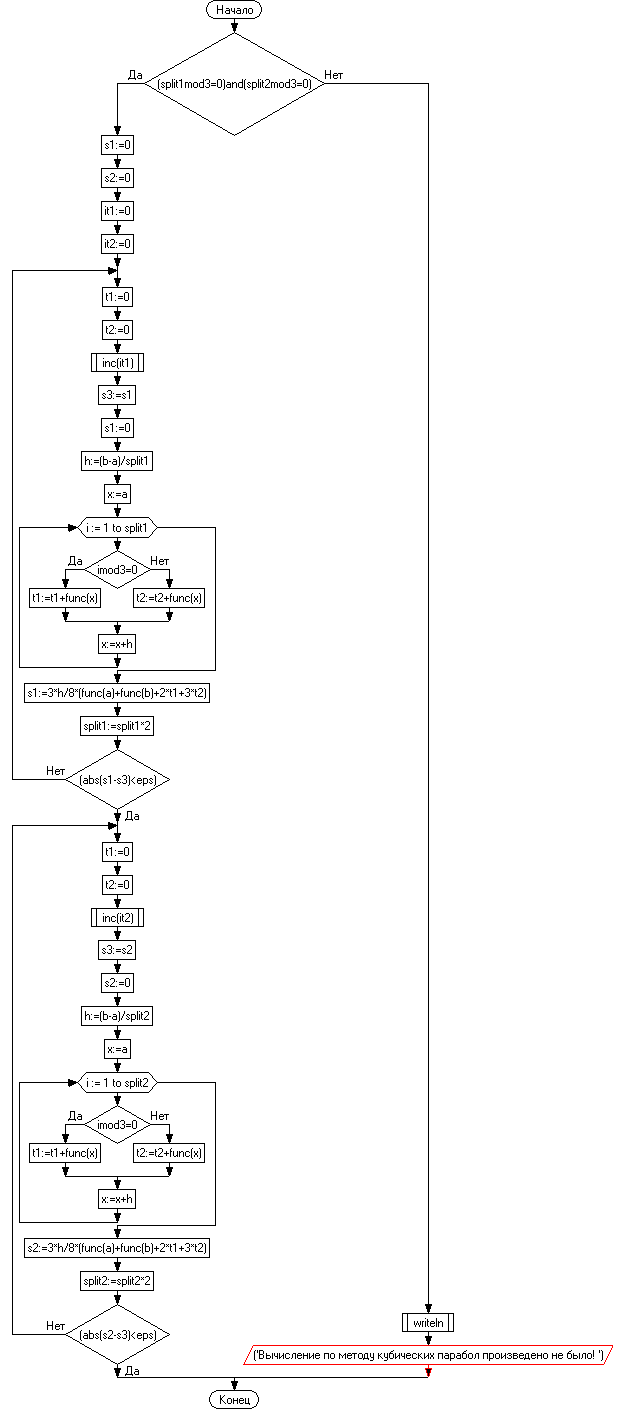
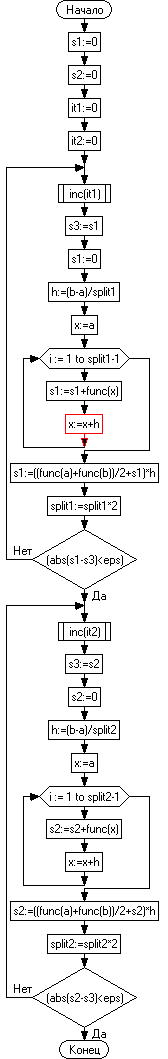
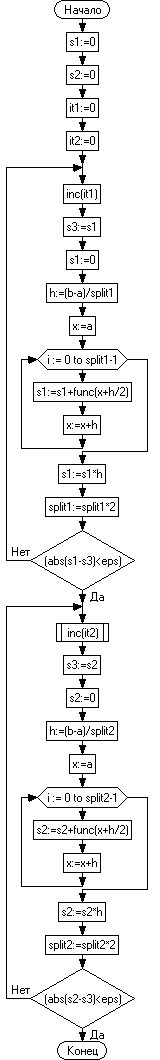
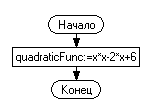
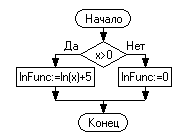
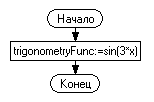
s3, t{1 .. 4) – промежуточные суммы из методов вычислений, тип extended (вещественные числа)

*Результат:*

split{1 .. 8} – получившиеся количества разбиений

it{1 .. 8} – количество итераций вычисления интеграла, тип longint (в случае больших чисел)

s{1 .. 8} – значения интеграла, вычисленные разными методами тип extended (в случае больших чисел)

**Блок-схемы**

trigonometryFunc

inputData

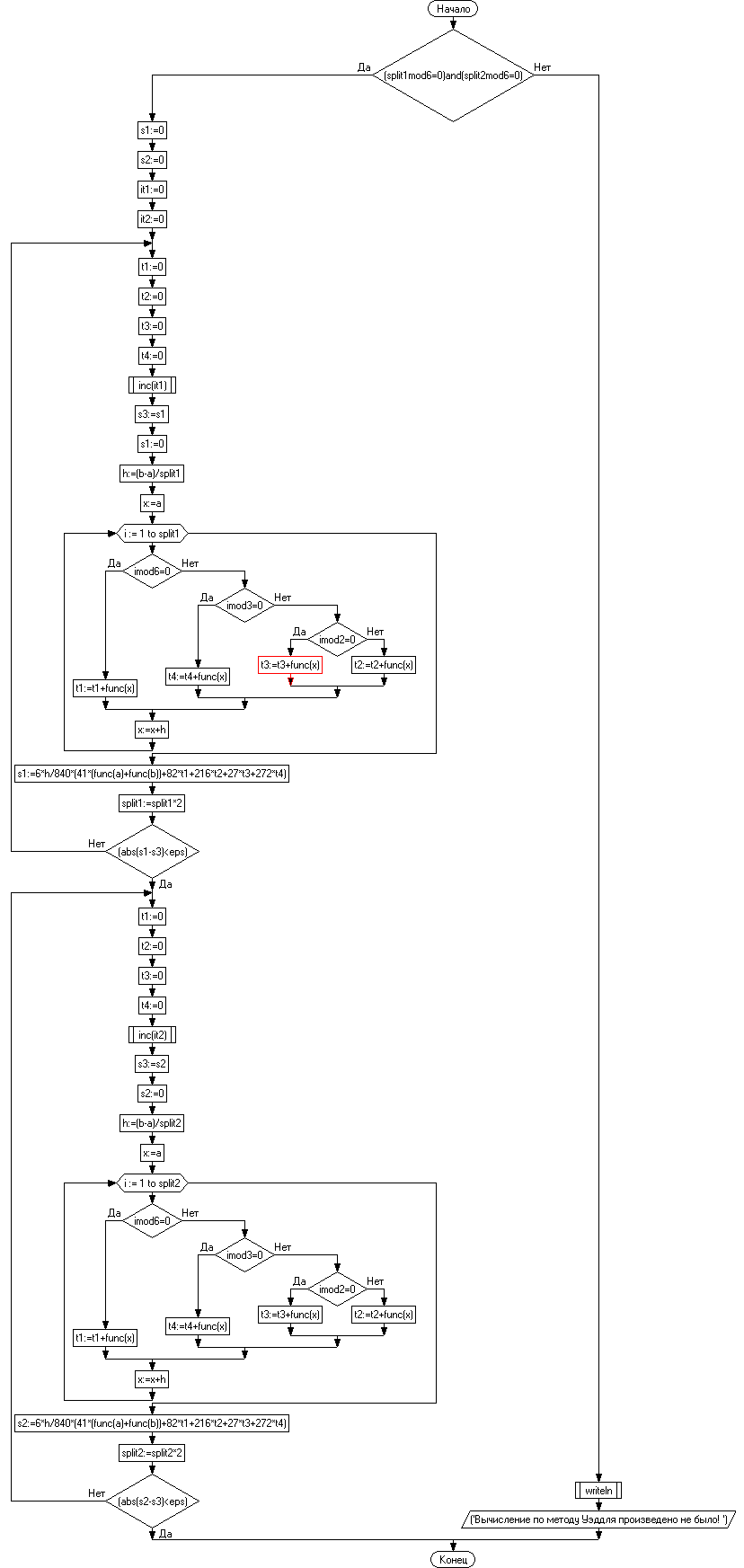
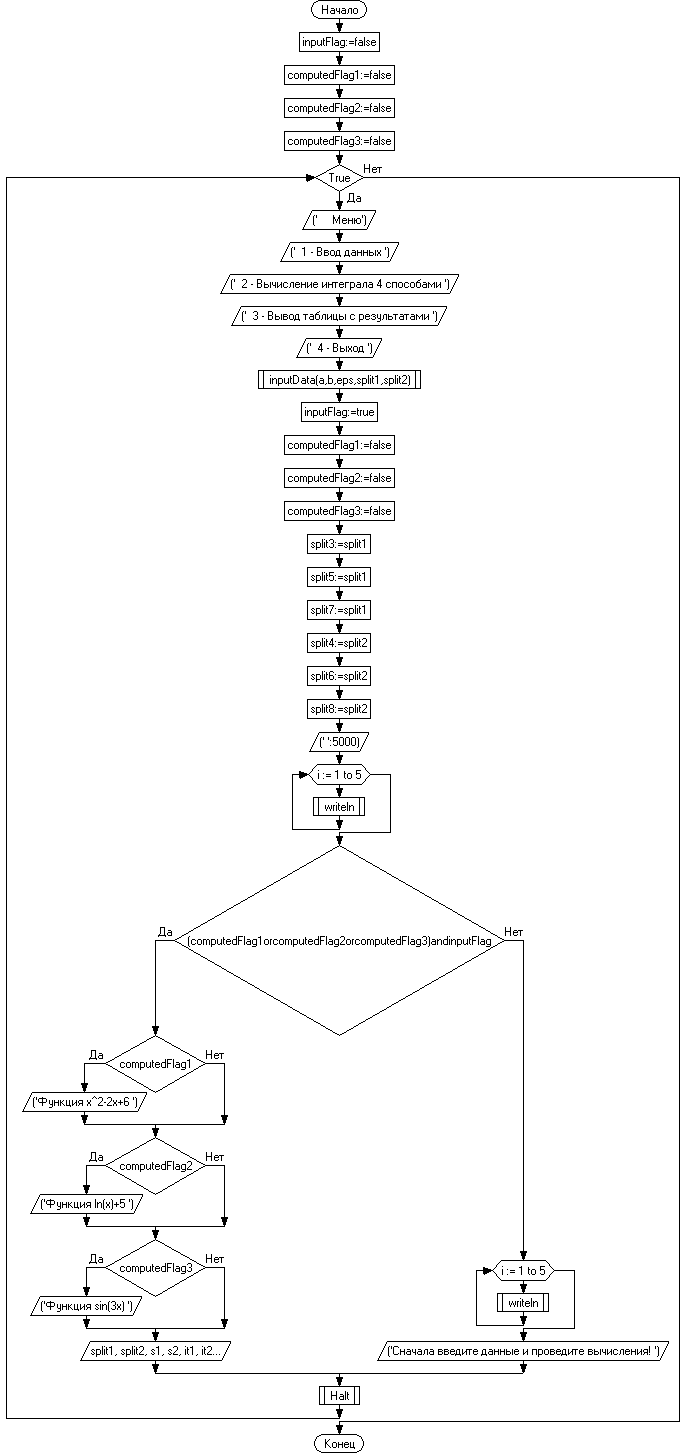
lnFunc

quadraticFunc

. ComputingMethod1

ComputingMethod3

ComputingMethod2



ComputingMethod4

**Листинг программы**

**Основная программа**

program mod\_4\_lab\_1;

{Моисеенко Богдан ФН12-11Б Вариант 17

Вычислить значение определенного интеграла подынтегральных функций на заданном

интервале, заданной точностью и 2 заданными начальными разбиениями 4 методами}

uses

procedure\_module in 'procedure\_module.pas',

SysUtils;

Var

answer : byte;

a, b, eps : double;

split1, split2, split3, split4, split5, split6, split7, split8, i : longint;

it1, it2, it3, it4, it5, it6, it7, it8 : longint;

x, s1, s2, s3, s4, s5, s6, s7, s8 : extended;

inputFlag, computedFlag1, computedFlag2, computedFlag3: boolean;

begin

inputFlag:=false;

computedFlag1:=false;

computedFlag2:=false;

computedFlag3:=false;

while True do

begin

write(' ': 2000); { очистка экрана }

writeln;

if inputFlag then writeln('Данные введены ');

if computedFlag1 then writeln('Вычисления проведены для первой функции : a = ', a:3:2, ', b = ', b:3:2, ', точность - ', eps:5:4)

else

if computedFlag2 then writeln('Вычисления проведены для второй функции : a = ', a:3:2, ', b = ', b:3:2, ', точность - ', eps:5:4)

else if

computedFlag3 then writeln('Вычисления проведены для третьей функции : a = ', a:3:2, ', b = ', b:3:2, ', точность - ', eps:5:4);

for i := 1 to 5 do writeln;

writeln(' Меню');

writeln(' 1 - Ввод данных ');

writeln(' 2 - Вычисление интеграла 4 способами ');

writeln(' 3 - Вывод таблицы с результатами ');

writeln(' 4 - Выход ');

for i := 1 to 5 do writeln;

write (' Введите пункт меню и нажмите <Enter>: ');

readln (answer);

case answer of

1: begin

write(' ': 5000); { очистка экрана }

for i := 1 to 5 do writeln;

inputData(a, b, eps, split1, split2);

inputFlag:=true;

computedFlag1:=false;

computedFlag2:=false;

computedFlag3:=false;

split3:=split1;

split5:=split1;

split7:=split1;

split4:=split2;

split6:=split2;

split8:=split2;

end;

2: begin

if inputFlag then

begin

write(' ': 5000);

for i := 1 to 5 do writeln;

writeln(' 2. Вычисление интеграла 4 способами.');

writeln(' 1 - Функция x^2-2x+6 ');

writeln(' 2 - Функция ln(x)+5 ');

writeln(' 3 - Функция sin(3x) ');

writeln(' 4 - Назад ');

for i := 1 to 5 do writeln;

write (' Введите пункт меню и нажмите <Enter>: ');

readln (answer);

case answer of

1: begin

write(' ': 5000);

computingMethod1(a, b, eps, split1, split2, it1, it2, s1, s2, quadraticFunc);

computingMethod2(a, b, eps, split3, split4, it3, it4, s3, s4, quadraticFunc);

computingMethod3(a, b, eps, split5, split6, it5, it6, s5, s6, quadraticFunc);

computingMethod4(a, b, eps, split7, split8, it7, it8, s7, s8, quadraticFunc);

computedFlag1:=true;

computedFlag2:=false;

computedFlag3:=false;

for i := 1 to 5 do writeln;

writeln('Вычисления проведены для 1 функции! ');

readln;

end;

2: begin

write(' ': 5000);

computingMethod1(a, b, eps, split1, split2, it1, it2, s1, s2, lnFunc);

computingMethod2(a, b, eps, split3, split4, it3, it4, s3, s4, lnFunc);

computingMethod3(a, b, eps, split5, split6, it5, it6, s5, s6, lnFunc);

computingMethod4(a, b, eps, split7, split8, it7, it8, s7, s8, lnFunc);

computedFlag2:=true;

computedFlag1:=false;

computedFlag3:=false;

for i := 1 to 5 do writeln;

writeln('Вычисления проведены для 2 функции! ');

readln;

end;

3: begin

write(' ': 5000);

computingMethod1(a, b, eps, split1, split2, it1, it2, s1, s2, trigonometryFunc);

computingMethod2(a, b, eps, split3, split4, it3, it4, s3, s4, trigonometryFunc);

computingMethod3(a, b, eps, split5, split6, it5, it6, s5, s6, trigonometryFunc);

computingMethod4(a, b, eps, split7, split8, it7, it8, s7, s8, trigonometryFunc);

computedFlag3:=true;

computedFlag1:=false;

computedFlag2:=false;

for i := 1 to 5 do writeln;

writeln('Вычисления проведены для 3 функции! ');

readln;

end;

4: begin

end

else

begin

write('Нет такого пункта. Нажмите <ENTER>. Повторите ввод. ');

readln

end;

end;

end

else

begin

write(' ': 5000);

for i := 1 to 5 do writeln;

writeln('Сначала введите данные! ');

readln;

end;

end;

3: begin

write(' ': 5000);

for i := 1 to 5 do writeln;

if (computedFlag1 or computedFlag2 or computedFlag3) and inputFlag then

begin

if computedFlag1 then writeln('Функция x^2-2x+6 ');

if computedFlag2 then writeln('Функция ln(x)+5 ');

if computedFlag3 then writeln('Функция sin(3x) ');

writeln;

writeln('Метод Количество Значение');

writeln(' разбиений итераций интеграла');

writeln('----------------------------------------------------------------------');

writeln('Центральных ', split1:5, ' ', it1:2, ' ', s1:8:5);

writeln('прямоугольников ', split2:5, ' ', it2:2, ' ', s2:8:5);

writeln('----------------------------------------------------------------------');

writeln('Трапеций ', split3:5, ' ', it3:2, ' ', s3:8:5);

writeln(' ', split4:5, ' ', it4:2, ' ', s4:8:5);

writeln('----------------------------------------------------------------------');

writeln('Кубических ', split5:5, ' ', it5:2, ' ', s5:8:5);

writeln('парабол ', split6:5, ' ', it6:2, ' ', s6:8:5);

writeln('----------------------------------------------------------------------');

writeln('Уэддля ', split7:5, ' ', it7:2, ' ', s7:8:5);

writeln(' ', split8:5, ' ', it8:2, ' ', s8:8:5);

readln;

end

else

begin

for i := 1 to 5 do writeln;

writeln('Сначала введите данные и проведите вычисления! ');

readln;

end;

end;

4: begin

Halt

end

else

begin

write('Нет такого пункта. Нажмите <ENTER>. Повторите ввод. ');

readln

end;

end;

end;

end.

**Модуль procedure\_module.pas**

unit procedure\_module;

interface

type

tFunc = function (x : extended):extended;

procedure inputData(var a, b, eps : double; var split1, split2 : integer);

procedure computingMethod1(a, b, eps : double; var split1, split2, it1, it2 : longint; var s1, s2: extended; func : tFunc);

procedure computingMethod2(a, b, eps : double; var split1, split2, it1, it2 : longint; var s1, s2: extended; func : tFunc);

procedure computingMethod3(a, b, eps : double; var split1, split2, it1, it2 : longint; var s1, s2: extended; func : tFunc);

procedure computingMethod4(a, b, eps : double; var split1, split2, it1, it2 : longint; var s1, s2: extended; func : tFunc);

function quadraticFunc(x : extended):extended;

function lnFunc(x : extended):extended;

function trigonometryFunc(x : extended):extended;

implementation

procedure inputData(var a, b, eps : double; var split1, split2 : integer);

begin

writeln(' 1. Ввод данных. ');

writeln('Для метода кубических парабол разбиение должно быть кратно 3,');

writeln('а для метода Уэддля - кратно 6. ');

writeln;

write('Введите начало интервала: ');

readln(a);

write('Введите конец интервала: ');

readln(b);

write('Введите точность вычислений: ');

readln(eps);

write('Введите начальное разбиение 1: ');

readln(split1);

write('Введите начальное разбиение 2: ');

readln(split2);

end;

procedure computingMethod1(a, b, eps : double; var split1, split2, it1, it2 : longint; var s1, s2: extended; func : tFunc);

var

i : longint;

h, s3, x : extended;

begin

s1:=0;

s2:=0;

it1:=0;

it2:=0;

repeat

inc(it1);

s3:=s1;

s1:=0;

h:=(b-a)/split1;

x:=a;

for i:=0 to split1-1 do begin

s1:=s1+func(x+h/2);

x:=x+h;

end;

s1:=s1\*h;

split1:=split1\*2;

until (abs(s1-s3)<eps);

repeat

inc(it2);

s3:=s2;

s2:=0;

h:=(b-a)/split2;

x:=a;

for i:=0 to split2-1 do begin

s2:=s2+func(x+h/2);

x:=x+h;

end;

s2:=s2\*h;

split2:=split2\*2;

until (abs(s2-s3)<eps);

end;

procedure computingMethod2(a, b, eps : double; var split1, split2, it1, it2 : longint; var s1, s2: extended; func : tFunc);

var

i : longint;

h, s3, x : double;

begin

s1:=0;

s2:=0;

it1:=0;

it2:=0;

repeat

inc(it1);

s3:=s1;

s1:=0;

h:=(b-a)/split1;

x:=a;

for i:=1 to split1-1 do begin

s1:=s1+func(x);

x:=x+h;

end;

s1:=((func(a)+func(b))/2+s1)\*h;

split1:=split1\*2;

until (abs(s1-s3)<eps);

repeat

inc(it2);

s3:=s2;

s2:=0;

h:=(b-a)/split2;

x:=a;

for i:=1 to split2-1 do begin

s2:=s2+func(x);

x:=x+h;

end;

s2:=((func(a)+func(b))/2+s2)\*h;

split2:=split2\*2;

until (abs(s2-s3)<eps);

end;

procedure computingMethod3(a, b, eps : double; var split1, split2, it1, it2 : longint; var s1, s2: extended; func : tFunc);

var

i : longint;

h, s3, x, t1, t2 : extended;

begin

if (split1 mod 3 = 0) and (split2 mod 3 = 0) then

begin

s1:=0;

s2:=0;

it1:=0;

it2:=0;

repeat

t1:=0;

t2:=0;

inc(it1);

s3:=s1;

s1:=0;

h:=(b-a)/split1;

x:=a;

for i:=1 to split1 do begin

if i mod 3 = 0 then

t1:=t1+func(x)

else

t2:=t2+func(x);

x:=x+h;

end;

s1:=3\*h/8\*(func(a)+func(b)+2\*t1+3\*t2);

split1:=split1\*2;

until (abs(s1-s3)<eps);

repeat

t1:=0;

t2:=0;

inc(it2);

s3:=s2;

s2:=0;

h:=(b-a)/split2;

x:=a;

for i:=1 to split2 do begin

if i mod 3 = 0 then

t1:=t1+func(x)

else

t2:=t2+func(x);

x:=x+h;

end;

s2:=3\*h/8\*(func(a)+func(b)+2\*t1+3\*t2);

split2:=split2\*2;

until (abs(s2-s3)<eps);

end

else

begin

writeln;

writeln('Вычисление по методу кубических парабол произведено не было! ');

end;

end;

procedure computingMethod4(a, b, eps : double; var split1, split2, it1, it2 : longint; var s1, s2: extended; func : tFunc);

var

i : longint;

h, s3, x, t1, t2, t3, t4 : extended;

begin

if (split1 mod 6 = 0) and (split2 mod 6 = 0) then

begin

s1:=0;

s2:=0;

it1:=0;

it2:=0;

repeat

t1:=0;

t2:=0;

t3:=0;

t4:=0;

inc(it1);

s3:=s1;

s1:=0;

h:=(b-a)/split1;

x:=a;

for i:=1 to split1 do begin

if i mod 6 = 0 then

t1:=t1+func(x)

else if i mod 3 = 0 then

t4:=t4+func(x)

else if i mod 2 = 0 then

t3:=t3+func(x)

else

t2:=t2+func(x);

x:=x+h;

end;

s1:=6\*h/840\*(41\*(func(a)+func(b))+82\*t1+216\*t2+27\*t3+272\*t4);

split1:=split1\*2;

until (abs(s1-s3)<eps);

repeat

t1:=0;

t2:=0;

t3:=0;

t4:=0;

inc(it2);

s3:=s2;

s2:=0;

h:=(b-a)/split2;

x:=a;

for i:=1 to split2 do begin

if i mod 6 = 0 then

t1:=t1+func(x)

else if i mod 3 = 0 then

t4:=t4+func(x)

else if i mod 2 = 0 then

t3:=t3+func(x)

else

t2:=t2+func(x);

x:=x+h;

end;

s2:=6\*h/840\*(41\*(func(a)+func(b))+82\*t1+216\*t2+27\*t3+272\*t4);

split2:=split2\*2;

until (abs(s2-s3)<eps);

end

else

begin

writeln;

writeln('Вычисление по методу Уэддля произведено не было! ');

end;

end;

function quadraticFunc(x : extended):extended;

begin

quadraticFunc:=x\*x-2\*x+6

end;

function lnFunc(x : extended):extended;

begin

if x>0 then

lnFunc:=ln(x)+5

else

lnFunc:=0

end;

function trigonometryFunc(x : extended):extended;

begin

trigonometryFunc:=sin(3\*x)

end;

end.

**Тестовые исходные данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Функция | Введенные данные | Вычисленное значение (все методы) | Точное значение |
|  | a = 0, b= 10, eps = 0.1,  split1 = 2, split2 = 3 | 293.31299  293.32031 | = 293. (3) |
| 293.23571  293.25523 |
| split не кратно 3 |
| split не кратно 6 |
| ln(x)+5 | a = -5, b= 10, eps = 0.01,  split1 = 3, split2 = 9 | 139.92189  139.91974 | 139.914645471 |
| 139.90787  139.90472 |
| 139.90533  139.91266 |
| split не кратно 6 |
| sin(3x) | a = -5, b= 10, eps = 0.001,  split1 = 3, split2 = 9 | 0.57685  0.57695 | 0.57671765472 |
| 0.57724  0.57741 |
| 0.57774  0.57740 |
| 0.57757  0.57729 |

**Скриншоты**

