**МИНОБРНАУКИ РОССИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ  
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»  
(БГТУ им. В.Г.Шухова)**

**Кафедра программного обеспечения вычислительной техники и автоматизированных систем**

Лабораторная работа №3

Дисциплина: Вычислительная математика

по теме «Численное интегрирование»

Выполнил: ст. группы ВТ-22  
Макаров Даниил Сергеевич

Проверила: Бондаренко Т.В.

**Белгород 2018**

**Лабораторная работа №3**

**Численное интегрирование**

**Цель работы:** изучить понятие квадратурной формулы; изучить основные способы численного интегрирования; получить практические навыки решения задачи численного интегрирования с помощью ЭВМ.

**Задания к работе**

1. Вычислить «вручную» интегралы из таблицы вариантов заданий:
   * точно (все 3 интеграла);
   * по формуле центральных (средних) прямоугольников, используя для оценки точности двойной просчёт при n1=8; n2=10 (интеграл 1);
   * по формуле трапеций при n=8 (интеграл 1 и 2);
   * по формуле парабол (Симпсона) при n=8 (интеграл 1 и 3).
2. Определить погрешность вычисления интеграла 1 по каждой из формул. Результаты представить в виде таблицы 3.1.
3. Описать в модуле функции, которые возвращают приближенные значения интегралов от функции f(x) с оценкой точности по принципу Рунге для методов центральных прямоугольников, трапеций и парабол. Исходными данными являются: подынтегральная функция f(x); пределы интегрирования a, b; начальное число отрезков разбиения n; точность вычисления ε. Необходимые для работы значения подынтегральной функции вычисляются непосредственной подстановкой значений аргумента в вычислительную формулу функции.
4. Составить программу для вычисления приближенных значений интегралов согласно варианту (все функции из таблицы вариантов заданий) с использованием всех функций, описанных в модуле.
5. Вычислить «вручную» интеграл из столбца 1 таблицы заданий по формуле Гаусса при n = 2. Значения узлов ti и весов Аi приведены в таблице 3.2.
6. Описать в модуле функцию для вычисления приближенного значения интеграла от функции f(x) по формуле Гаусса при n = 1, 2, 3 ,4. Добавить в составленную программу вычисление приближенных значений 25 интегралов (все функции из таблицы вариантов заданий) с использованием формулы Гаусса.
7. Заполнить значения погрешности вычисления интеграла 1 в таблице 3.1 для формулы Гаусса при n = 1, 2, 3, 4.

Таблица 3.1 Оценка погрешности вычислений

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Погрешность | Формула центр. прямоугольников | | Формула трапеций n=8 | Формула парабол  n=8 | Формула Гаусса | | | |
| n=8 | n=10 | n=1 | n=2 | n=3 | n=4 |
| Δ |  |  |  |  |  |  |  |  |
| δ |  |  |  |  |  |  |  |  |

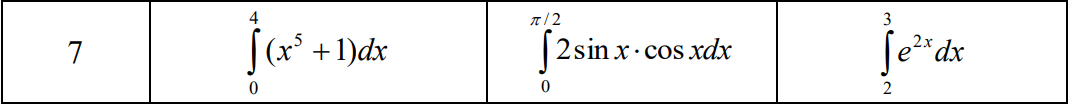
Таблица 3.2. Значение узлов и весов для формулы Гаусса

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| n | 1 | 2 | 3 | 4 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Содержание отчета**

1. Титульный лист.
2. Цель работы. Вариант задания.
3. Текст задания к работе.
4. Выполнение задания соответствующего варианта вручную полностью. Все действия выполняемые при решении задачи вручную расписывать подробно с указанием всех промежуточных операций. Указание только окончательного ответа не допускается.
5. Текст программы, включающий необходимые комментарии и спецификации подпрограмм.
6. Результаты работы программы.

Вариант 7



Вычисление значений интеграла вручную

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № | | 1 | | 2 | | 3 | |
| f(x)0 | | x^5+1 | | 2sin(x)\*cos(x) | | e^2x | |
| Нижний предел | | 0 | | 0 | | 2 | |
| Верхний предел | | 4 | | 1,570796327 | | 3 | |
| Точное значение | | 686,666 | | 1 | | 174,42 | |
| Интеграл 1 | | | | | | | | | | | | |
| n | | 8 | | 10 | | 2 | |  | |  |  |  |
| h | | 0,5 | | 0,4 | | 2 | |  | |  |  |  |
|  | |  | |  | |  | |  | |  |  |  |
| Формула центральных прямоугольников | | | | | | | | | | | | |
| n=8 | | xi | | x i+1/2 | | y i+1/2 | | n=10 | | xi | x i+1/2 | y i+1/2 |
| 0 | | 0 | | 0,25 | | 1,0009766 | | 0 | | 0 | 0,2 | 1,00032 |
| 1 | | 0,5 | | 0,75 | | 1,2373047 | | 1 | | 0,4 | 0,6 | 1,07776 |
| 2 | | 1 | | 1,25 | | 4,0517578 | | 2 | | 0,8 | 1 | 2 |
| 3 | | 1,5 | | 1,75 | | 17,413086 | | 3 | | 1,2 | 1,4 | 6,37824 |
| 4 | | 2 | | 2,25 | | 58,665039 | | 4 | | 1,6 | 1,8 | 19,89568 |
| 5 | | 2,5 | | 2,75 | | 158,27637 | | 5 | | 2 | 2,2 | 52,53632 |
| 6 | | 3 | | 3,25 | | 363,59082 | | 6 | | 2,4 | 2,6 | 119,8138 |
| 7 | | 3,5 | | 3,75 | | 742,57715 | | 7 | | 2,8 | 3 | 244 |
| Результат | | 673,40625 | | | | | | 8 | | 3,2 | 3,4 | 455,3542 |
|  | |  | |  | |  | | 9 | | 3,6 | 3,8 | 793,3517 |
|  | |  | |  | |  | | Результат | | 678,1632 | | |
|  | |  | |  | |  | |  | |  |  |  |
| Формула трапеции | | | | | |  | | Формула парабол | | | |  |
| n=8 | | xi | | yi | |  | | n=8 | | xi | yi |  |
| 0 | | 0 | | 1 | |  | | 0 | | 0 | 1 |  |
| 1 | | 0,5 | | 1,03125 | |  | | 1 | | 0,5 | 1,03125 |  |
| 2 | | 1 | | 2 | |  | | 2 | | 1 | 2 |  |
| 3 | | 1,5 | | 8,59375 | |  | | 3 | | 1,5 | 8,59375 |  |
| 4 | | 2 | | 33 | |  | | 4 | | 2 | 33 |  |
| 5 | | 2,5 | | 98,65625 | |  | | 5 | | 2,5 | 98,65625 |  |
| 6 | | 3 | | 244 | |  | | 6 | | 3 | 244 |  |
| 7 | | 3,5 | | 526,21875 | |  | | 7 | | 3,5 | 526,2188 |  |
| 8 | | 4 | | 1025 | |  | | 8 | | 4 | 1025 |  |
| Результат | | 713,25 | | | |  | | Результат | | 687 | |  |
|  | |  | |  | |  | |  | |  |  |  |
| Формула Гауcса | | | | | | | | | | |  |  |
| n=2 | | x | | y | | ti | | Ai | | y\*a[i] |  |  |
| 0 | | 0,84529946 | | 1,431571703 | | -0,57735 | | 1 | | 1,431572 |  |  |
| 1 | | 3,15470054 | | 313,4573172 | | 0,5773503 | | 1 | | 313,4573 |  |  |
| Результат | | 629,7777778 | | | | | | | | |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Погрешность | Формула центральных прямоугольников | | Формула трапеций n=8 | Формула парабол n=8 | Формула Гаусса |
| n=8 | n=10 | n=2 |
| ∆ | 13,25975 | 8,5028 | 26,584 | 0,334 | 56,88822 |
| ∂ | 1,93% | 1,24% | 3,87% | 0,05% | 8,28% |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Интеграл 2 | | | | |
| n | 8 | | |  |
| h | 0,19634954 | | |  |
|  |  | | |  |
| Формула трапеции | | | | |
| n=8 | xi | | | yi |
| 0 | 0 | | | 1 |
| 1 | 0,19634954 | | | 1,370965924 |
| 2 | 0,39269908 | | | 1,689246397 |
| 3 | 0,58904862 | | | 1,942610078 |
| 4 | 0,78539816 | | | 2,121320344 |
| 5 | 0,9817477 | | | 2,218509458 |
| 6 | 1,17809725 | | | 2,230442497 |
| 7 | 1,37444679 | | | 2,156660883 |
| 8 | 1,57079633 | | | 2 |
| Результат | 2,990355516 | | | |
| Интеграл 3 | | | | |
| n | | 8 |  | |
| h | | 0,125 |  | |
|  | |  |  | |
| Формула парабол | | | | |
| n=8 | | xi | yi | |
| 0 | | 2 | 54,59815003 | |
| 1 | | 2,125 | 70,10541235 | |
| 2 | | 2,25 | 90,0171313 | |
| 3 | | 2,375 | 115,5842845 | |
| 4 | | 2,5 | 148,4131591 | |
| 5 | | 2,625 | 190,5662685 | |
| 6 | | 2,75 | 244,6919323 | |
| 7 | | 2,875 | 314,1906603 | |
| 8 | | 3 | 403,4287935 | |
| Результат | | 174,4190788 | | |

**Заголовочный файл integral.h**

#ifndef INTEGRAL\_H\_

#define INTEGRAL\_H\_

double (\*integral\_func) (double);

double integral\_rect(double (\*func) (double),double a,double b,int n,double eps);

double integral\_tr(double (\*func) (double),double a,double b,int n,double eps);

double integral\_pb(double (\*func) (double),double a,double b,int n,double eps);

double integral\_gause(double (\*func) (double),double a,double b,int n);

#endif

**Исходный код integral.h**

#include "integral.h"

#include <stdlib.h>

#include <stdio.h>

double integral\_rect(double (\*func) (double),double a,double b,int n,double eps){

double o = 1.0/3.0;

double result,prev\_result,result\_sub,sum=0,x;

int count = n;

double step=(b-a)/(float)count;

for(int i=0;i<count;i++){

x=a+(i+0.5)\*step;

sum+=func(x);

}

prev\_result=sum\*step;

result\_sub=1;

while(result\_sub>=eps){

count\*=2;

step=(b-a)/(float)count;

sum=0;

for(int i=0;i<count;i++){

x=a+(i+0.5)\*step;

sum+=func(x);

}

result=step\*sum;

result\_sub=o\*abs(result-prev\_result);

prev\_result=result;

}

return result;

}

double integral\_tr(double (\*func) (double),double a,double b,int n,double eps){

double o = 1.0/3.0;

double result,prev\_result,result\_sub,sum=0,x;

int count = n;

double step=(b-a)/(float)count;

for(int i=0;i<count;i++){

x=a+i\*step;

sum+=func(x);

}

prev\_result=step\*(((func(a)+func(b))/2.0)+sum);

result\_sub=1;

while(result\_sub>=eps){

count\*=2;

step=(b-a)/(float)count;

sum=0;

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

x=a+i\*step;

sum+=func(x);

}

result=step\*(((func(a)+func(b))/2.0)+sum);

result\_sub=o\*abs(result-prev\_result);

prev\_result=result;

}

return result;

}

double integral\_pb(double (\*func) (double),double a,double b,int n,double eps){

double o = 1.0/15.0;

double result,prev\_result,result\_sub,sum\_1=0,sum\_2=0,x;

int count = n;

double step=(b-a)/(float)count;

//цикл для суммы нечетных

for(int i=1;i<(count-1);i=i+2){

x=a+i\*step;

sum\_1+=func(x);

}

//цикл для суммы четных

for(int i=2;i<(count-2);i=i+2){

x=a+i\*step;

sum\_2+=func(x);

}

prev\_result=step/3.0\*(func(a)+func(b)+4\*sum\_1+2\*sum\_2);

result\_sub=1;

while(result\_sub>=eps){

count\*=2;

step=(b-a)/(float)count;

sum\_1=0;

sum\_2=0;

//цикл для суммы нечетных

for(int i=1;i<(count-1);i=i+2){

x=a+i\*step;

sum\_1+=func(x);

}

//цикл для суммы четных

for(int i=2;i<(count-2);i=i+2){

x=a+i\*step;

sum\_2+=func(x);

}

result=step/3.0\*(func(a)+func(b)+4\*sum\_1+2\*sum\_2);

result\_sub=o\*abs(result-prev\_result);

prev\_result=result;

}

return result;

}

double integral\_gause(double (\*func) (double),double a,double b,int n){

double gause\_scale[4][4]={

{2.0,0.0,0.0,0.0},

{1.0,1.0,0.0,0.0},

{0.5555555556,0.8888888888,0.5555555556,0.0},

{0.3478548451,0.6521451549,0.6521451549,0.3478548451}

};

double gause\_node[4][4]={

{0.0,0.0,0.0,0.0},

{-0.5773502692,0.5773502692,0.0,0.0},

{-0.7745966692,0.0,0.7745966692,0.0},

{-0.8611363115,-0.3399810436,0.3399810436,0.8611363115}

};

double result,step,sum=0;

step=(b-a)/(float)n;

result=(b-a)/2.0;

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

sum+=gause\_scale[n-1][i]\*func((b+a)/2.0+((b-a)/2.0)\*gause\_node[n-1][i]);

}

result\*=sum;

return result;

}

**Исходный код main.c**

#include <stdlib.h>

#include <stdio.h>

#include "integral.h"

#include <math.h>

double func\_1 (double x){

return (x\*x\*x\*x\*x)+1;

}

double func\_2 (double x){

return 2\*sin(x)+cos(x);

}

double func\_3 (double x){

return exp(2\*x);

}

int main(){

double result;

integral\_func=func\_1;

result=integral\_rect(integral\_func,0,4,4,0.000001);

printf("%f\n",result);

result=integral\_tr(integral\_func,0,4,4,0.000001);

printf("%f\n",result);

result=integral\_pb(integral\_func,0,4,4,0.000001);

printf("%f\n",result);

result=integral\_gause(integral\_func,0,4,3);

printf("%f\n",result);

return 0;

}

