МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Нижегородский государственный технический университет

им. Р.Е. Алексеева» (НГТУ)

**Лабораторная работа №1**

**«Элементарная теория погрешностей»**

Выполнил:

студент группы 21-САИ

Краличев И. Е.

Проверил:

Катаева Л. Ю.

Нижний Новгород,

2023 г.

**Вариант №10**

**Постановка задачи:**

**Задание №1**

Определить, какое равенство точнее.



**Задание №2**

Округлить сомнительные цифры числа, оставив верные знаки: а) в узком смысле; б) в широком смысле.

Определить абсолютную погрешность результата.



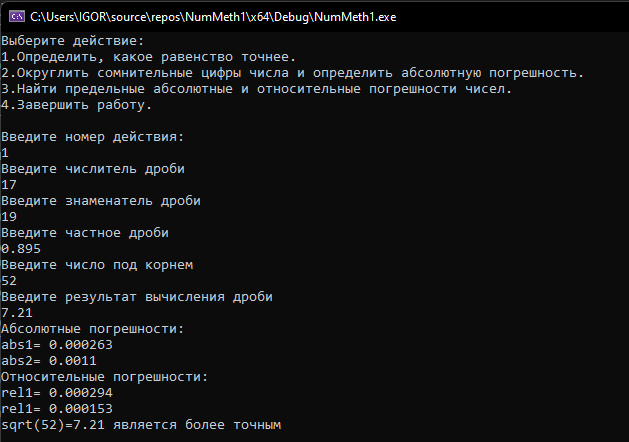
**Задание №3**

Найти предельные абсолютные и относительные погрешности чисел, если они имеют только верные цифры: а) в узком смысле; б) в широком смысле.



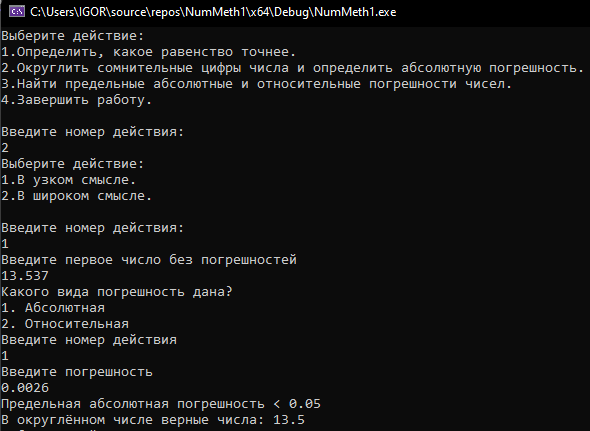
**Результаты выполнения программ.**

**Задание №1**

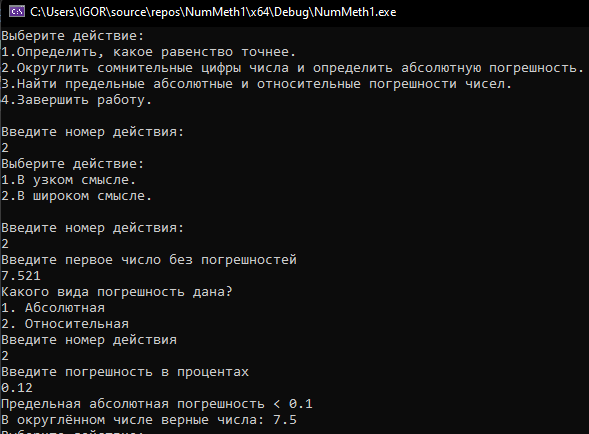
****

**Задание №2**

**А).**

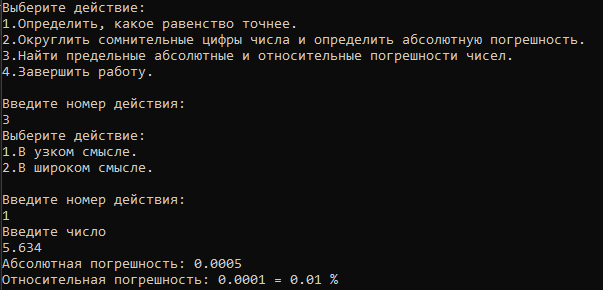
****

**Б).**

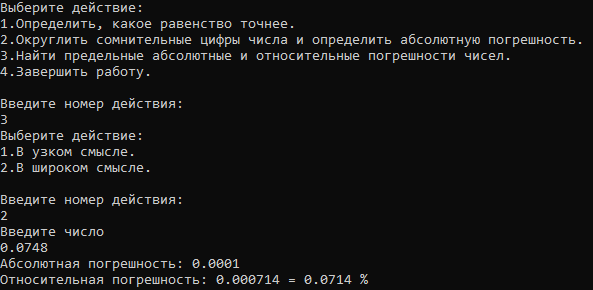


**Задание №3**

**А).**

****

**Б).**

****

**Код реализации на С++**

#include <iostream>

#include <string>

#include <iomanip>

#include <math.h>

using namespace std;

// Проверка на ввод числа

void checker(double a) {

if (cin.fail()) {

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Вы ввели не число!\n" << endl;

}

}

// Округление числа

double corrNum(double num, double numAfterSepar, bool choice) {

double degreeOfRound = pow(10, numAfterSepar); // для степени округления числа

double roundNum; // округлённое число

if (choice) {

roundNum = round(num \* degreeOfRound) / degreeOfRound;

}

else {

if (num > 0)

roundNum = floor(num \* degreeOfRound) / degreeOfRound;

else

roundNum = ceil(num \* degreeOfRound) / degreeOfRound;

}

return roundNum;

}

//Задание 1

void firstTask() {

double numerator, denominator, privateDiv, numRoot, eqRoot, absDiv, absRoot, relDiv, relRoot;

double mass[4]; // массив с равенствами (0,2- левая часть; 1,3- правая часть)

// Ввод дроби

do {

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Введите числитель дроби \n";

cin >> numerator;

} while (cin.fail());

do {

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Введите знаменатель дроби \n";

cin >> denominator;

} while (cin.fail());

do {

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Введите частное дроби \n";

cin >> privateDiv;

} while (cin.fail());

// Запись в массив

mass[0] = numerator / denominator;

mass[1] = privateDiv;

// Ввод числа под коорнем

do {

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Введите число под корнем \n";

cin >> numRoot;

} while (cin.fail());

do {

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "Введите результат вычисления дроби \n";

cin >> eqRoot;

} while (cin.fail());

// Запись в массив

mass[2] = sqrt(numRoot);

mass[3] = eqRoot;

absDiv = abs(mass[0] - mass[1]);

absRoot = abs(mass[2] - mass[3]);

cout << setprecision(3) << "Абсолютные погрешности: \n" << "abs1= " << absDiv << "\n" << "abs2= " << absRoot << "\n";

relDiv = absDiv / mass[1];

relRoot = absRoot / mass[3];

cout << setprecision(3) << "Относительные погрешности: \n" << "rel1= " << relDiv << "\n" << "rel1= " << relRoot << "\n";

if (relDiv < relRoot) {

cout << numerator << "/" << denominator << "=" << mass[1] << " является более точным\n" << endl;

}

else if (relRoot < relDiv) {

cout << "sqrt(" << numRoot << ")" << "=" << mass[3] << " является более точным\n" << endl;

}

else { cout << "Одинаковая точность\n"; };

}

//Задание 2

void secondTask(double num, int choice) {

int choiceErr;

double absErr, relErr, roundNum, absNext, numAfterSepar, ultimateAbs;

switch (choice) {

//Узкий смысл

case 1:

cout << "Какого вида погрешность дана? \n";

cout << "1. Абсолютная \n" << "2. Относительная\n";

cout << "Введите номер действия \n";

cin >> choiceErr;

// Правильность выбора метода

while (!(choiceErr == 1 || choiceErr == 2)) {

cout << "Выберите 1 или 2!\n";

cin >> choiceErr;

//Проверка на число

checker(choiceErr);

}

switch (choiceErr) {

//Абсолютная погрешность

case 1:

cout << "Введите погрешность \n";

cin >> absErr;

numAfterSepar = int(log10(1 / absErr));

roundNum = corrNum(num, numAfterSepar, choice == 1);

absNext = abs(roundNum - num) + absErr;

ultimateAbs = 5 \* pow(10, -1 - int(log10(1 / absErr)));

if (ultimateAbs < absErr) { ultimateAbs = ultimateAbs \* 10; }

while (absNext > ultimateAbs) {

numAfterSepar = numAfterSepar - 1;

roundNum = corrNum(roundNum, numAfterSepar, choice == 1);

absNext = abs(roundNum - num) + absErr;

ultimateAbs = ultimateAbs \* 10;

}

cout << "Предельная абсолютная погрешность < " << ultimateAbs << endl;

cout << "В округлённом числе верные числа: " << roundNum << endl;

break;

//Относительная погрешность

case 2:

cout << "Введите погрешность в процентах \n";

cin >> relErr;

checker(relErr);

relErr = relErr / 100;

absErr = num \* relErr;

numAfterSepar = int(log10(1 / absErr));

roundNum = corrNum(num, numAfterSepar, choice == 1);

absNext = abs(roundNum - num) + absErr;

ultimateAbs = 5 \* pow(10, -1 - int(log10(1 / absErr)));

if (ultimateAbs < absErr) { ultimateAbs = ultimateAbs \* 10; }

while (absNext > ultimateAbs) {

numAfterSepar = numAfterSepar - 1;

roundNum = corrNum(roundNum, numAfterSepar, choice == 1);

absNext = abs(roundNum - num) + absErr;

ultimateAbs = ultimateAbs \* 10;

}

cout << "Предельная абсолютная погрешность < " << ultimateAbs << endl;

cout << "В округлённом числе верные числа: " << roundNum << endl;

break;

}

break;

//Широкий смысл

case 2:

cout << "Какого вида погрешность дана? \n";

cout << "1. Абсолютная \n" << "2. Относительная\n";

cout << "Введите номер действия \n";

cin >> choiceErr;

while (!(choiceErr == 1 || choiceErr == 2)) {

cout << "Выберите 1 или 2!\n";

cin >> choiceErr;

//Проверка на число

checker(choiceErr);

}

switch (choiceErr) {

//Абсолютная погрешность

case 1:

cout << "Введите погрешность \n";

cin >> absErr;

checker(absErr);

numAfterSepar = int(log10(1 / absErr));

roundNum = corrNum(num, numAfterSepar, choice == 1);

absNext = abs(roundNum - num) + absErr;

ultimateAbs = 1 \* pow(10, -int(log10(1 / absErr)));

while (absNext > ultimateAbs) { // 5 \* pow(10, - 1 - int(log10(1 / ab1)))

numAfterSepar = numAfterSepar - 1;

roundNum = corrNum(roundNum, numAfterSepar, choice == 1);

absNext = abs(roundNum - num) + absErr;

ultimateAbs = ultimateAbs \* 10;

}

cout << "Предельная абсолютная погрешность < " << ultimateAbs << endl;

cout << "В округлённом числе верные числа: " << roundNum << endl;

break;

//Относительная погрешность

case 2:

cout << "Введите погрешность в процентах \n";

cin >> relErr;

checker(relErr);

relErr = relErr / 100;

absErr = num \* relErr;

numAfterSepar = int(log10(1 / absErr));

roundNum = corrNum(num, numAfterSepar, choice == 1);

absNext = abs(roundNum - num) + absErr;

ultimateAbs = 1 \* pow(10, -int(log10(1 / absErr)));

while (absNext > ultimateAbs) {

numAfterSepar = numAfterSepar - 1;

roundNum = corrNum(roundNum, numAfterSepar, choice == 1);

absNext = abs(roundNum - num) + absErr;

ultimateAbs = ultimateAbs \* 10;

}

cout << "Предельная абсолютная погрешность < " << ultimateAbs << endl;

cout << "В округлённом числе верные числа: " << roundNum << endl;

break;

}

break;

}

}

// Задание 3

void thirdTask(int& signFigure, int& numOfSignFigure, int& numOfZeros) {

cout << "Введите число \n";

char mass[100];

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cin.getline(mass, 100);

char separator = '.';

numOfZeros = 0;

numOfSignFigure = 0;

bool keySeparToSign = false;

bool check = false; // проверки

int i = 0;

string s;

while (mass[i] != '\0') {

if (mass[i] != separator) {

s += mass[i];

while (numOfSignFigure != 0) {

numOfSignFigure++; break;

}

if ((mass[i] != '0') && (check == false)) {

signFigure = mass[i] - '0';

check = true;

numOfSignFigure = 1;

}

if (keySeparToSign == true) {

numOfZeros++;

}

}

else {

s.clear();

keySeparToSign = true;

}

i++;

}

}

int main()

{

setlocale(LC\_ALL, "RUS");

int n = 1; //Для выхода из программы

int choiceMain; //Выбор действия

int choiceVar; //В каком смысле (узкий/широкий)

while (n == 1) {

cout << "Выберите действие:\n";

cout << "1.Определить, какое равенство точнее.\n";

cout << "2.Округлить сомнительные цифры числа и определить абсолютную погрешность.\n";

cout << "3.Найти предельные абсолютные и относительные погрешности чисел.\n";

cout << "4.Завершить работу.\n\n";

cout << "Введите номер действия:\n";

cin >> choiceMain;

//Проверка на число

checker(choiceMain);

//Главная выборки

switch (choiceMain) {

//Задание 1

case 1:

firstTask();

break;

//Задание 2

case 2:

cout << "Выберите действие:\n";

cout << "1.В узком смысле.\n";

cout << "2.В широком смысле.\n\n";

cout << "Введите номер действия:\n";

cin >> choiceVar;

// Правильность выбора метода

while (!(choiceVar == 1 || choiceVar == 2)) {

cout << "Выберите 1 или 2!\n";

cin >> choiceVar;

//Проверка на число

checker(choiceVar);

}

cout << "Введите первое число без погрешностей\n";

double num;

cin >> num;

checker(num);

//Уточнение выборки

switch (choiceVar) {

// Узкий смысл

case 1:

secondTask(num, choiceVar);

break;

// Широкий смысл

case 2:

secondTask(num, choiceVar);

break;

}

break;

//Задание 3

case 3:

cout << "Выберите действие:\n";

cout << "1.В узком смысле.\n";

cout << "2.В широком смысле.\n\n";

cout << "Введите номер действия:\n";

cin >> choiceVar;

int signFigure, numOfSignFigure, numOfZeros; //для вычисления

double absErr, relErr; //погрешности

// Правильность выбора метода

while (!(choiceVar == 1 || choiceVar == 2)) {

cout << "Выберите 1 или 2!\n";

cin >> choiceVar;

//Проверка на число

checker(choiceVar);

}

switch (choiceVar) {

// Узкий смысл

case 1:

thirdTask(signFigure, numOfSignFigure, numOfZeros);

absErr = 5 \* pow(10, -numOfZeros - 1);

cout << "Абсолютная погрешность: " << absErr << endl;

relErr = 1 / (2 \* signFigure \* pow(10, numOfSignFigure - 1));

cout << "Относительная погрешность: " << relErr << " = " << relErr \* 100 << " %" << endl;

break;

// Широкий смысл

case 2:

thirdTask(signFigure, numOfSignFigure, numOfZeros);

absErr = 1 \* pow(10, -numOfZeros);

cout << "Абсолютная погрешность: " << absErr << endl;

relErr = 1 / (2 \* signFigure \* pow(10, numOfSignFigure - 1));

cout << "Относительная погрешность: " << relErr << " = " << relErr \* 100 << " %" << endl;

break;

}

break;

// Завершить работу

case 4:

cout << "Завершение работы...\n\n";

n = 0;

break;

}

}

}