**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра САПР**

**ОТЧЁТ**

**По практической работе №1**

**По дисциплине «Программирование»**

Студент гр. 3351 \_\_\_\_\_\_\_\_\_\_\_\_ Морозов А.А.

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_ Рыжов Н.Г.

Санкт-Петербург

2024

цель работы

1. Написать программу, оценивающую значение машинного нуля X0, и машинного эпсилон *ε*M для разных типов данных: float, double. Для нахождения машинного нуля возьмем любое положительное число A, например, А=1, и будем делить его на 2 до тех пор, пока A > 0. Последнее положительное число и будет машинным нулем X0 для выбранного типа данных (точнее, любое число, по модулю меньшее данной величины). Для определения *ε*M возьмем два числа: A = 1 и *ε*= A. Будем делить *ε* на 2 столько раз, пока результат деления *ε*, сложенный с A, будет больше A. Последнее *ε,* при котором A+*ε* >1 и будет *ε*M. Для каждого типа данных вывести значение машинного нуля, машинного эпсилон и числа итераций k (число делений на 2), необходимых соответственно для вычисления X0 и *ε*M. Важно: при вычислении выражений производится автоматическое преобразование типов, чтобы привести операнды выражений к общему типу или чтобы тип меньшей разрядности расширился (повысился) до большего совместимого типа. При выполнении арифметических преобразований операнды типа float преобразуются к типу double.

2. Используя алгоритм вычисления *ε*M провести эксперименты по определению числа итераций k при различных начальных значениях D, обращая внимание на малые значения (D<10-35). Дополните отчет о работе разделом «Результаты экспериментов», в котором приведите график зависимости k от D для типа float, укажите причину изменения числа итераций k при малых значениях D и постоянства k для остальных значений D.

**ТЕКСТ ПРОГРАММЫ**

#include <iostream>

using namespace std;

void MachineNullFloat(float A) {

int k = 0;

float res;

while (A > 0) {

res = A;

A /= 2;

k += 1;

}

cout << "Машинный ноль [float] - " << res << ".\n";

cout << "Количество итераций - " << k << ".\n";

}

float testGraph(float A) {

int k = 0;

float res;

while (A > 0) {

res = A;

A /= 2;

k += 1;

}

return res;

}

void MachineNullDouble(double A) {

int k = 0;

double res;

while (A > 0) {

res = A;

A /= 2;

k += 1;

}

cout << "Машинный ноль [double] - " << res << ".\n";

cout << "Количество итераций - " << k << ".\n";

}

void MachineEpsilonFloat(float A, float B) {

int k = 0;

float res;

while (A + B > 1) {

res = B;

B /= 2;

k += 1;

}

cout << "Машинный эпсилон [float] - " << res << ".\n";

cout << "Количество итераций - " << k << ".\n";

}

void MachineEpsilonDouble(double A, double B) {

int k = 0;

float res;

while (A + B > 1) {

res = B;

B /= 2;

k += 1;

}

cout << "Машинный эпсилон [double] - " << res << ".\n";

cout << "Количество итераций - " << k << ".\n";

}

int main() {

setlocale(LC\_ALL, "RU");

float A\_float, B\_float, d = 1000000.0f, epsilon = d, k = 0;

double A\_double, B\_double;

cin >> A\_float;

cin >> A\_double;

B\_float = A\_float;

B\_double = A\_double;

MachineNullFloat(A\_float);

MachineNullDouble(A\_double);

MachineEpsilonFloat(A\_float, B\_float);

MachineEpsilonDouble(A\_double, B\_double);

cout << "Значения для исследования и построения графика.\n";

for (d; d >= testGraph(A\_float); d /= 2) {

epsilon = d;

k = 0;

while (d + epsilon > d) {

epsilon /= 2;

k++;

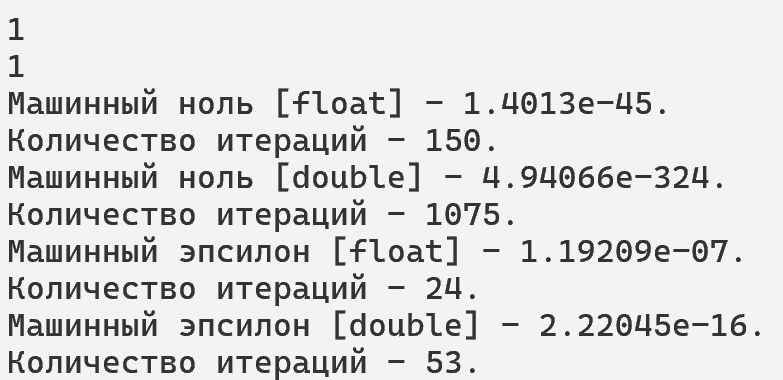
}

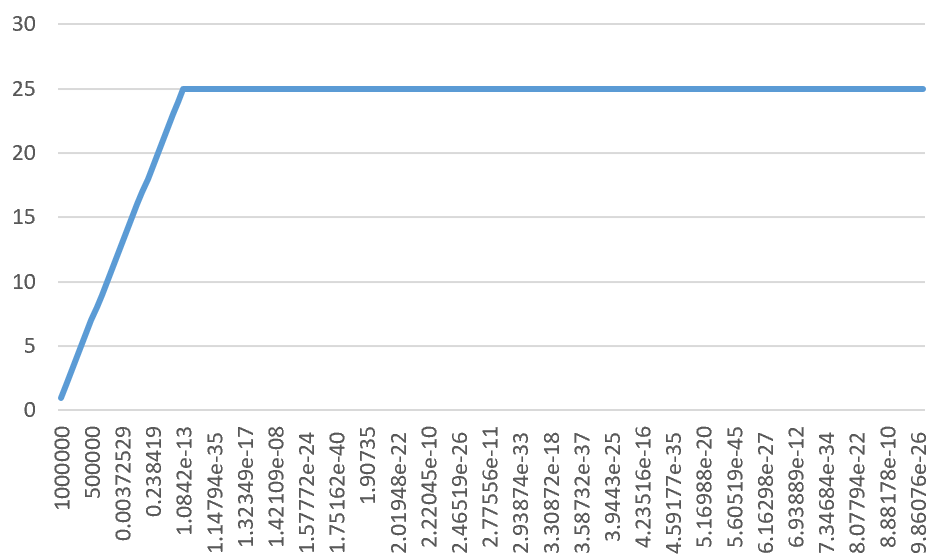
cout << d << " " << k << endl;

}

}

**Результаты выполнения программы**





**Пути дальнейшего улучшения программы**

1. Улучшение визуальной части программы
2. Повышение точности вычислений