МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ

по практической работе №1

по дисциплине «Вычислительная математика»

Тема: Особенность машинной арифметики, точность вычисления на ЭВМ

Студент гр. 8304 Николаева М. А.

Преподаватель Попова Е. В.

Санкт-Петербург

2019

Вариант 10.

Цель работы.

Изучить особенности вычислений с плавающей точкой.

Основные теоретические положения.

В фундаменте математического анализа прочно утвердилась система действительных чисел. Однако, как бы она не упрощала анализ, практические вычисления вынуждены обходиться без нее.

Обычным способом аппроксимации системы действительных чисел в ЭВМ посредством конкретных математических представлений являются числа с плавающей точкой. Множество F чисел с плавающей точкой характеризуется четырьмя параметрами: основанием b, точностью t и интервалом показателей [L,M]. Каждое число с плавающей точкой, принадлежащее F, имеет значение

$$x = \pm\left(\frac{d_1}{b} + \frac{d_2}{b^2} + \cdots + \frac{d_t}{b^t}\right) b^n,$$

где целые числа $d_1, d_2, \ldots, d_t$ удовлетворяют неравенствам$0 \leqslant d_j < b$ $(j = \overline{1,t})$ $L \leqslant n \leqslant M$. Если для каждого ненулевого x из F справедливо $d_1 \neq 0$, то система F называется нормализованной. Целое число n называется показателем, а число $f = \sum_{j=1}^{t} d_j / b^j$ – дробной частью. Обычно целое число b_n хранится по той или иной схеме представления, принятой для целых чисел, например, величины со знаком, дополнения до единицы или дополнения до двух. Если принять $-N \leqslant n < N$, где $N = 2^{m-1}$ то переходим к общепринятой терминологии, при которой t – разрядность мантиссы, m – разрядность порядка.

Действительная машинная реализация представлений чисел с плавающей точкой может отличатся в деталях от рассматриваемой идеальной, однако различия несущественны, и на практике их почти всегда можно игнорировать, анализируя основные проблемы ошибок округления. Величина b^{-t} является оценкой относительной точности плавающей арифметики, которая характеризуется посредством машинного эпсилон, т.е. наименьшего числа с плавающей точкой ε,

такого, что 1+ε>1. Точное значение машинного эпсилон зависит не только то указанных выше параметров, но и от принятого способа округления.

В вычислительных машинах используются различные системы чисел с плавающей точкой, причем в некоторых ЭВМ несколько систем. Так, для современных ПЭВМ характерно применение двух систем, которые называются обычной точностью и удвоенной точностью.

Рассматриваемое множество F не является континуумом или даже бесконечным множеством. Оно содержит ровно $2(b-1)b^t(M-L+1)+1$ чисел, которые расположены неравномерно (равномерность расположения имеет место лишь при фиксированном показателе). В силу того, что F – конечное множество, не представляется возможным сколь-нибудь детально отобразить континуум действительных чисел. Например, действительные числа модулей, большим максимального элемента из F, вообще не могут быть отображены, причем последнее справедливо также в отношении ненулевых действительных чисел, меньших по абсолютной величине по сравнению с наименьшим положительным числом из F, и, наконец, каждое число из F должно представлять целый интервал действительных чисел, для которой, как и для любой модели, присущи допущения и ограничения.

На множестве F определены арифметические операции в соответствии с тем, как они выполняются ЭВМ. Эти операции, в свою очередь моделируются в машине посредством приближений, называемых плавающими операциями. Для плавающих операций сложения, вычитания, умножения и деления существует возможность возникновения ошибок округления, переполнения и появления машинного нуля. Следует отметить, что операции плавающего сложения и умножения коммутативны, но не ассоциативны, и дистрибутивный закон для них также не выполняется. Невыполнение указанных алгебраических законов, имеющих фундаментальное значение для математического анализа, приводит к сложности анализа плавающих вычислений и возникающих при этом ошибок.

Постановка задачи.

Используя ряд специально разработанных программ, выполнить исследования машинной арифметики и точности вычислений на ПЭВМ. Порядок выполнения работы следующий:

1) Исследование распределения нормализованных чисел с плавающей точкой на вещественной оси для различных значений параметров b, m, t.

2) Вычисление значения величины машинного эпсилон при различных значениях константы ***c.***

3) Исследование абсолютных и относительных ошибок округления при вычислениях с плавающей точкой сумм чисел при различных значениях шага суммирования

4) Исследование проявления ошибок округления, возникающих при вычислении показательной функции e^x , для чисел с плавающей точкой для двух вариантов алгоритма вычислений, а также скорости сходимости обоих вариантов

5) Исследование округления Truncate.

Выполнение работы.

1. Проведены исследования распределения нормализованных чисел с плавающей точкой на вещественной оси для различных значений параметров b, m, t. Результаты расчетов см. в табл. 1.

Таблица 1 — числа, сгенерированные программой с разными значениями параметров b, m, t.

	b = 2, t = 3, m = 1
0	0.000000
1	0.250000
2	0.312500

3	0.375000
4	0.437500
5	0.500000
6	0.625000
7	0.750000
8	0.875000
9	1.000000
10	1.250000
11	1.500000
12	1.750000

Распределение нормализованных чисел с плавающей точкой на вещественной оси неравномерно. Плотность распределения увеличивается при движении к границе диапазона.

2. Были вычислены значения ε, при разных значениях аргумента c. Результаты вычислений см. в табл. 2.

Таблица 2 — результаты вычисления ε при разных значениях c

Значение *c*	Значение ε
7	$43*10^{-19}$
8	$86*10^{-19}$
9	$86*10^{-19}$

При маленьком значении *c* значение ε небольшое. Увеличение происходит в виде геометрической прогрессии со знаменателем 2, т.е. с увеличением *c* в два раза ε так же увеличивается в 2 раза (с точностью до 10^{-19}). График зависимости ε от *c* показан на рис. 1. Для построения графика были использованы дополнительные значения *c* и соответствующие им значения *e*.

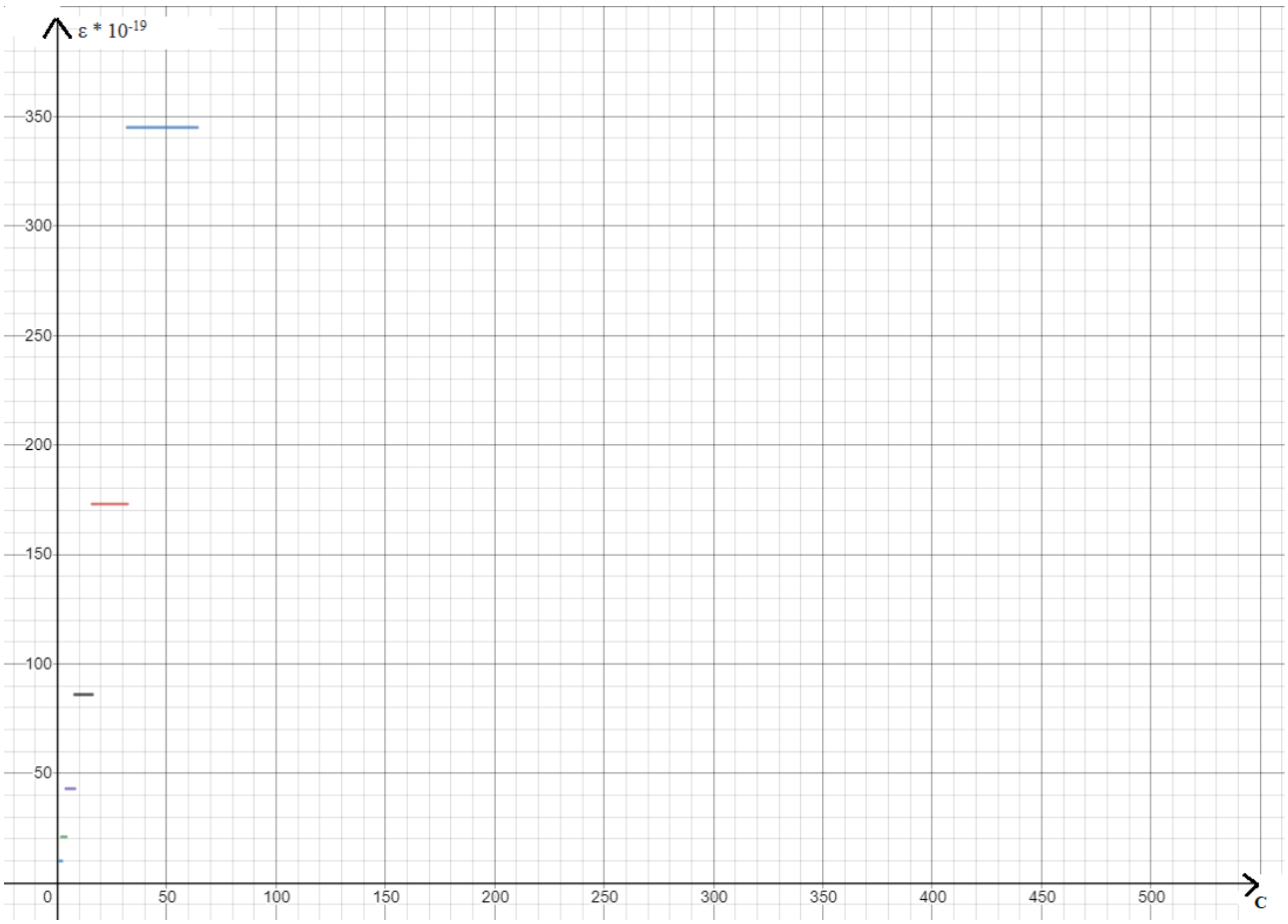

Рисунок 1 — график зависимости ε от c

3. Было проведено исследование абсолютных и относительных ошибок округления при вычислениях с плавающей точкой сумм чисел при различных значениях шага суммирования. Результаты вычислений см. в табл. 3.

Таблица 3 — результаты исследования абсолютных и относительных ошибок округления (N – шаг суммирования x – dx – абсолютная погрешность, (x-dx)/x – относительная погрешность)

N	x - dx	(x – dx)/x
9	0.0000000596	0.000006%
97	0.0000000307	0.000003%
456	0.0000000987	0.000010%
863	0.0000000268	0.000003%
1290	0.0000000037	0.000000%
1498	0.0000000441	0.000004%

Для каждого N абсолютная погрешность увеличивалась с шагами в ходе суммирования (происходило накопление ошибки), а относительная ошибка была постоянной (абсолютная ошибка накапливалась равномерно).

4. Было проведено исследование проявления ошибок округления, возникающих при вычислении показательной функции e^x для чисел с плавающей точкой для двух вариантов алгоритма вычислений, а также найдены скорости сходимости обоих вариантов. Результаты обработки программой введенных данных см. в табл. 4.

Таблица 4 — исследование проявления ошибок округления, возникающих при вычислении функции e^x для двух алгоритмов.

Введенное значение x	Введенное значение ε	Разложение Тейлора	Улучшенный алгоритм	Абсолютная погрешность	Относительная погрешность
9	0.001	8103.08370349120742029 итераций	8103.08392757537968001 итерация	0.000224084172260	0.000003 %
11	0.001	59874.14132550389330034 итерация	59874.14171519778030001 итерация	0.0003896938869733	0.000001%
31	0.001	2904884966524742190000000000088 итераций	2904884966524737500000000000001 итерация	0.046875000000000	0.000000%

При увеличении аргумента абсолютная погрешность возрастает, в то время как относительная погрешность мала и при больших значениях аргумента выходит за пределы машинного эпсилон, следовательно, не может быть посчитана. Сходимость ряда Тейлора вычисляется медленно. Так, для аргумента 31 и порядка 0.001 требуется 88 итераций, каждая из которых по объему вычислений превышает предыдущую. Улучшенный алгоритм является более рациональным вариантом, так как на целых числах дает сходимость за 1 итерацию.

5. Было проведено исследование округления функцией Truncate. Результаты см. в табл. 5.

Таблица 5 – округление с помощью Truncate.

	Значение до округления	После округления
0	0.14	0
1	5.7	5
2	1.2	1
3	-3.6	-3
4	0.98	0
5	1.45	1
6	-0.5	0

Метод Truncate округляет числа к ближайшему целому числу в сторону нуля. Исходный код программы для исследования Truncate:

```
using System;

class Program
{
    static void Main() {
        int n = 7;
        double[] a = new double[7] {0.14, 5.7, 1.2, -3.6, 0.98, 1.45, -0.5};

        for (int i = 0; i < n; i++){
            Console.Write("Значение до округления "+a[i]+ " после округления
"+Math.Truncate(a[i])+'\n');
        }
    }
}
```

Выводы.

В ходе выполнения заданий лабораторной работы, были исследованы машинная арифметика, точность вычислений на ПЭВМ, распределение нормализованных чисел на вещественной оси, абсолютные и относительные ошибки округления при вычислениях с плавающей точкой, зависимость машинного эпсилон от значения константы и округление чисел с помощью метода Truncate. Все результаты исследований были занесены в таблицы, для некоторых из них был построен график.