



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

---

Факультет «Информатика и системы управления»

Кафедра «Программное обеспечение ЭВМ и информационные технологии»

**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

**«ДЛИННАЯ АРИФМЕТИКА»**

по курсу «Типы и структуры данных»

**Вариант 1**

Студент: Писаренко Дмитрий Павлович

Группа: ИУ7-34Б

Студент

\_\_\_\_\_

Писаренко Д.П.

*подпись, дата*

*фамилия, и.о.*

Преподаватель

\_\_\_\_\_

Барышникова М.Ю.

*подпись, дата*

*фамилия, и.о.*

## Условие задачи

Составить программу умножения или деления двух чисел, где порядок имеет до 5 знаков: от  $-99999$  до  $+99999$ , а мантисса – до 30 знаков. Программа должна осуществлять ввод чисел и выдавать либо верный результат в указанном формате (при корректных данных), либо сообщение о невозможности произвести счет.

Смоделировать операцию деления действительного числа в форме  $\pm m.n \text{ E } \pm K$ , где суммарная длина мантиссы ( $m+n$ ) – до 30 значащих цифр, а величина порядка  $K$  - до 5 цифр, на целое число длиной до 30 десятичных цифр. Результат выдать в форме  $\pm 0.m1 \text{ E } \pm K1$ , где  $m1$  - до 30 значащих цифр, а  $K1$  - до 5 цифр.

# Техническое задание

## Входные данные

Действительное число вида  $\pm m.nE\pm K$ . Суммарная длина строки – до 30 значащих цифр ( $m + n$ ) + символ точки и знака. Величина порядка  $K$  – 5 цифр и символ знака.

Целое число вида  $\pm d$ . Длина до 30 значащих цифр.

```
typedef struct
{
    char sign;
    int mantissa[MANTISSA_LEN + 1];
    int order;
} float_t;

typedef struct
{
    char sign;
    int digits[MANTISSA_LEN];
} integer_t;
```

```
char sign - знак числа
int mantissa[MANTISSA_LEN + 1] - массив цифр действительного числа
int order - порядок действительного числа
int digits[MANTISSA_LEN] - массив цифр целого числа
```

Описание полей структур `float_t` и `integer_t`

## Ограничения на входные данные

- Действительное число не более 30 значащих цифр
- Целое число не более 30 значащих цифр
- Порядок не более 5 значащих цифр
- Нельзя вводить пробелы до/после знаков, до порядка

## Выходные данные

Действительное число вида  $\pm 0.m1E\pm K1$ ,  $m1$  – до 30 значащих цифр,  $K1$  – до 5 значащих цифр.

## Способ обращения к программе

```
./build_release.sh
./app.exe
```

## Алгоритм программы

1. Производится ввод действительного числа посимвольно через `getchar()`. В функции ввода действительного числа на блоки разделены: ввод знака, ввод мантиссы, ввод `E` или `e`, ввод знака с порядком. При некорректном вводе какого-либо блока программа завершается с соответствующим ненулевым кодом возврата и выводит сообщение об ошибке на экран.

2. Происходит нормализация действительного числа.

3. Производится ввод целого числа посимвольно через `getchar()`. В функции ввода целого числа на блоки разделены: ввод знака, ввод цифр. При некорректном вводе какого-либо блока программа завершается с соответствующим ненулевым кодом возврата и выводит сообщение об ошибке на экран. Также в этой функции проверяется деление на ноль.

4. Производится деление действительного числа на целое. Сначала в промежуточной функции `pre_division` считается, сколько раз можно вычесть из первого числа второе, затем происходит деление в столбик первого числа на второе.

5. Производится округление, и, соответственно, нормализация результата, если длина мантиссы стала равна 31 знаку.

6. Результат выводится на экран.

# Тестирование

## Позитивные тесты

#	Входные данные	Выходные данные
1	+123.123E+123 +5351	+0.23009344047841524948607736E+122
2	-123.52 +11	-0.11229090909090909090909090909E+2
3	+2 +2	+0.1E+1
4	+100 -10	-0.1E+2
5	+252.32E+99999 +1321	+0.191006813020439061317183951E+99999
6	+752.32E+99999 +25124	+0.29944276389110014328928514E+99998
7	-321 +321	-0.1E+1
8	-15123.123E+15 +251	-0.602514860557768924302788844E+17
9	+0000123.123E+123 +5351	+0.23009344047841524948607736E+122
10	+123.123E+000123 +5351	+0.23009344047841524948607736E+122
11	+0.0 -12412	-0.0E0
12	+1123.411E+52 +512	+0.2194162109375E+53

### Негативные тесты

#	Входные данные	Выходные данные	Результат
1	122.123	Ошибка ввода: вначале не был введен знак	Код ошибки 2
2	+123 +123.12	Ошибка ввода: были введены не цифры	Код ошибки 5
3	\n	Пустой ввод	Код ошибки 1
4	+122.12E+FEffrr	Ошибка ввода: в порядке были введены не цифры	Код ошибки 6
5	+11111111111111111111 11111111111111111111	Произошло переполнение	Код ошибки 3
6	+123.123.123	Ошибка ввода: было введено более одной точки	Код ошибки 4
7	+512.325E+150000	Порядок больше критического значения	Код ошибки 7
8	+512.41E-953 +0	Деление на ноль невозможно	Код ошибки 8

## Контрольные вопросы

*1. Каков возможный диапазон чисел, представляемых в ПК?*

На это влияет тип чисел, размер выделенной для их хранения памяти и разрядность машины. Если переменная типа unsigned integer занимает 4 байта (32 бита), то диапазон значений будет от 0 до  $2^{32}-1$ .

*2. Какова возможная точность представления чисел?*

Это зависит от количества памяти, выделенной для хранения мантиссы. В среднем под хранение выделяют 16-30 разрядов.

*3. Какие стандартные операции возможны над числами?*

Сложение, вычитание, умножение, деление, взятия остатка, сравнение.

*4. Какой тип данных может выбрать программист, если обрабатываемые числа превышают возможный диапазон представления чисел в ПК?*

Структура.

*5. Как можно осуществить операции над числами, выходящими за рамки машинного представления?*

Путем последовательного выполнения операций над всеми цифрами, начиная с конца массива.

## Вывод

В процессе выполнения лабораторной работы были изучены принципы работы длинной арифметики. Я узнал, как работать с числами, которые выходят за возможный диапазон значений каких-либо типов, представленных в ЯП.