

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

2023 г.

# Описание условия задачи

Смоделировать операцию умножения действительного числа на действительное число в форме  $\pm m.n \text{ E } \pm K$ , где суммарная длина мантиссы первого сомножителя ( $m+n$ ) - до 35 значащих цифр, второго – до 40 значащих цифр, а величина порядка  $K$  - до 5 цифр. Результат выдать в форме  $\pm 0.m1 \text{ E } \pm K1$ , где  $m1$  – до 40 значащих цифр, а  $K1$  - до 5 цифр.

## Техническое задание

### Исходные данные:

На вход программе подаются два действительных числа в формате  $[+-]?m.n\text{E}[\pm]?k$  в отдельных строках, где в первом числе длина части до  $e$  не превышает 35 значащих цифр без учета точки и знаков  $+$  и  $-$ , а во втором случае – до 40 значащих цифр. В обоих случаях длина части после  $k$  не превышает 5 цифр без учета знаков  $+$  или  $-$ .

### Выходные данные:

Программа выдает действительное число, полученное перемножением полученных на входе чисел в формате  $[+-]0.m\text{E}[\pm]k$ , где длина  $m$  не превышает 40 цифр, а длина  $k$  - 5 цифр.

### Описание задания:

Перемножение двух действительных чисел, превышающих размер стандартных типов данных в языке C.

### Способы обращения к программе:

Запуск программы через терминал, в котором как ключи указываются перемножаемые числа, сначала первое, затем второе.

## Аварийные ситуации:

- 1) Недопустимые символы в первом числе или некорректный формат первого числа. Код ошибки – 1;
- 2) Недопустимые символы во втором числе или некорректный формат второго числа. Код ошибки – 2;
- 3) Превышение размера мантиссы у первого числа(>35 значащих цифр). Код ошибки – 3;
- 4) Превышение размера мантиссы у второго числа(>40 значащих цифр). Код ошибки – 4;
- 5) Превышение размера порядка у первого числа(>5 значащих цифр). Код ошибки – 5;
- 6) Превышение размера порядка у второго числа(>5 значащих цифр). Код ошибки – 6;
- 7) Подано только одно число или не поданы числа вовсе. Код ошибки – 7;
- 8) Переполнение размера порядка для результата умножения(>5 значащих цифр). Код ошибки – 8;

## Описание структур данных

```
struct long_float
{
    char mant_sign;
    int mantiss[MAX_MANTISS_SIZE];
    int order;
    size_t size;
};
```

Структура первого числа для умножения

- Mant\_sign – Знак мантиссы, если 1 – то число положительное, иначе отрицательное;
- Mantiss – Мантисса числа, хранится в виде массива целых, при этом в каждом из целых хранится одна цифра числа.
- Order – Порядок числа.
- Size – нынешний размер мантисс, то есть количество значащих цифр в норм. форме

## Описание алгоритма

1. Считать строки с числами из аргументов командной строки;
2. Перевести строки в структуры длинного числа;
3. Занулить все значения в переменной результата;
4. Провести операцию умножения чисел по следующему алгоритму:
  - a. Создать зануленный массив целых двойного размера для сохранения результата умножения;
  - b. Поочередно перемножить цифры мантиссы двух чисел в столбик (Результат умножения  $i$ -й цифры 1-го числа на  $j$ -ю второго записывается в  $i+j$ -ю результирующего);
  - c. Если массив получился длиннее максимальной длины мантиссы, то округлить ее, сделав сдвиг значений массива влево, на разницу в размере;
  - d. Сложить порядки чисел и вычесть из суммы единицу, это порядок результат умножения;
  - e. Провести над знаками мантисс чисел НЕ-исключающее-ИЛИ – знак результата;
  - f. Записать все значения в переменную результата;
5. Проверить длину получившегося порядка;
6. Вывести получившееся число.

## Тестовые данные

Позитивные тесты			
№	Описание	Вход	Выход
1	Простое число без точки и e	1231 1	+0.1231E+4
2	Простое число без точки	1231E10 1	+0.1231E+14
3	Простое число без e	1234.123 1	+0.1234123E+4
4	Простое число	1234.123E-4 1	+0.1234123E+0
5	Есть пустые нули до точки	12300E2 1	+0.123E+7
6	Есть пустые нули после точки	123120.13000E1 1	+0.12312013E+7
7	Есть пустые нули и до, и после точки	12341000.000E2 1	+0.12341E+10
8	Нули перед числом	000123.21E1 1	+0.12321E+4
9	+ Перед числом	+123.21E21 1	+0.12321E+24
10	- Перед числом	-123.21E21 1	-0.12321E+24
11	Пустые нули перед порядком	123.123E00123 1	+0.123123E+126
12	- Перед порядком	1.2E-31 1	+0.12E-30
13	Умножение на 2	1.2E1 2	+0.24E+2
14	Умножение на 10	1.2E+1 10	+0.12E+3
15	Умножение действительных чисел	1.2E3 102.412136E54	+0.1228945632E+60
16	Умножение чисел 35 знаков на 40	1...1E1 1...12E1	+0.1234567901234567901 234567901234567888889 E+51

17	Умножение чисел разных знаков	-123.123 123.213E32	-0.15170354199E+37
18	Умножение отрицательных чисел	-123.123 -123.213E32	0.15170354199E+37
<b>Негативные тесты</b>			
1	Пустой ввод		Not Enough Arguments.
2	Превышение длины мантиссы	1111111111111111 1111111111111111 1111.1 1	Too long mantiss.
3	Превышение длины порядка	1.2E111111 1	Too long order.
4	Буквы вместо числа	sqe 1	Incorrect format of one of numbers.
5	Буквы в числе	1sa.21 1	Incorrect format of one of numbers.
6	Две точки в числе	12.123.12E12 1	Incorrect format of one of numbers.
7	Точка в порядке	12.2E12.2 1	Incorrect format of one of numbers.
8	Нет мантиссы	E123 1	Can't find one of mantisses.
9	Выход за диапазон рез. порядка	1E9999 1E2	Too big result order.

## Ответы на вопросы

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

Диапазон чисел зависит от выбранного типа данных и разрядности процессора, например для 64-разрядного максимально возможное значение беззнакового числа равно 18 446 744 073 709 551 615.

2. *Какова возможная точность представления чисел, чем она определяется?*

Точность числа определяется количеством памяти, выделяемой под её мантиссу, например для типа `double` количество выделяемой памяти равно 52 бита, что позволяет хранить мантиссу до 4503599627370496.

3. *Какие стандартные операции возможны над числами?*

Зависит от типа переменной, например над целым типом возможны операции сравнения, сложения, вычитания, деления, умножения, взятия остатка, бинарные сдвиги.

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

В большинстве языков не реализованы структуры для чисел превышающих диапазон представления в ПК, поэтому для их обработки можно использовать либо массив символов, либо самописные конструкции, например структуры в СИ

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

Так как нет стандартных структур для чисел превышающих представление в ПК, то и стандартные операции над ними также отсутствуют, поэтому программисту вручную придется прописывать их, например методом деления или умножения столбиком.

## **Выводы**

В ходе работы я научился методам работы с числами, выходящими за стандартный диапазон значений и реализовал свою вариацию длинной арифметики.