



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 _____

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

«ДЛИННАЯ АРИФМЕТИКА»

Студент _____ Шимшир Эмирджан Османович _____
фамилия, имя, отчество

Группа _____ ИУ7-33Б _____

2022 г.

Оглавление

Техническое задание.....	3
Условие задачи.....	3
Входные данные.....	3
Выходные данные.....	3
Задача, реализуемая программой.....	3
Способ обращения к программе.....	3
Возможные аварийные ситуации и ошибки пользователя.....	4
Описание внутренних структур данных.....	4
Тесты.....	5
Контрольные вопросы.....	7
Вывод.....	8

Техническое задание

Условие задачи

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

Входные данные

1. Строка, в которой записано длинное целое число. Строка должна иметь вид $\pm n1$, где длина $n1$ не превышает 30 при этом ввод знака «+» необязателен.

2. Строка, в которой записано длинное вещественное число в экспоненциальной или обычной форме. Строка должна иметь вид $\pm m.n \text{ E } \pm K$, где суммарная длина мантииссы ($m+n$) – до 30 значащих цифр, а величина порядка K – до 5 цифр при этом ввод точки, экспоненты, пробелов и знака «+» необязателен.

Выходные данные

Строка, являющаяся результат операции умножения двух введенных чисел и представленная в виде $\pm 0.m1 \text{ E } \pm K1$, где $m1$ – до 30 значащих цифр, а $K1$ – до 5 цифр.

Задача, реализуемая программой

Осуществление операции умножение длинного целого числа на длинное вещественное число.

Способ обращения к программе

Запускается через терминал. Вводится вещественное число, затем целое.

Возможные аварийные ситуации и ошибки пользователя

Некорректный ввод (использование недопустимого символа, превышение максимально возможного количества символов, неверный формат ввода).

Переполнение порядка в результате умножения.

Во всех аварийных ситуациях в консоль выводятся соответствующие вспомогательные сообщения.

Описание внутренних структур данных

В программе используются две структуры данных: одна – для представления длинного вещественного числа – `double_t`, другая – для представления длинного целого числа – `int_t`.

typedef struct

```
{  
    char sign;  
    char num[MAX_DOUBLE_LEN];  
    short int len_num;  
    short int point_place;  
    int order;  
} double_t;
```

Структура `double_t` имеет 5 полей:

char sign – знак вещественного числа;

char num[MAX_DOUBLE_LEN] – массив символов для хранения мантиссы числа;

short int len_num – длина мантиссы;

short int point_place – положение точки отсчитывая от старшего разряда;

int order – порядок вещественного числа.

В данную структуру записывается введенное вещественное число и результат вычислений.

typedef struct

```
{  
    char num[MAX_INT_LEN + 1];  
    short int len_num;  
} int_t;
```

Структура int_t имеет два поля:

char num[MAX_INT_LEN + 1] – массив символов, представляющих цифры числа и знак, хранящийся в элементе с индексом 0.

short int len_num – длина целого числа.

В данную структуру записывается введенное целое число.

Описание алгоритма

Для осуществления операции умножения используется алгоритм вычисления методом Транхенберга. Он представляет собой модифицированный способ умножения «столбиком», в котором каждая цифра числа вычисляется сразу, что не требует отдельного хранения результатов умножения каждой цифры второго числа на первое.

Тесты

№	Что проверяется	Вещественное число	Целое число	Результат
1	Некорректный формат ввода	ert	--	Ошибка при чтении порядка
2	Некорректный формат ввода	iu7	--	При вводе использован некорректный символ
3	Некорректный формат ввода	1e23	FGH	При вводе использован некорректный символ
4	Некорректный формат ввода	-12.3e+12.3	--	Ошибка при чтении порядка
5	Некорректный	--1e+21	--	При вводе

	формат ввода			использован некорректный символ
6	Некорректный формат ввода	1e88	-1e56	При вводе использован некорректный символ
7	Больше 30 цифр в мантиссе	999999999999999 999999999999999 1	--	Превышено максимальное количество символов
8	Больше 5 цифр в порядке	1e999991	--	Ошибка при чтении порядка
9	Больше 30 цифр в целом числе	1e1	999999999999999 999999999999999 9999991	Превышено максимальное количество символов
10	Наибольшее значение	999999999999999 999999999999999 e99969	1	+0.999999999999999 999999999999999 999e99999
11	Наименьшее значение	-999999999999999 999999999999999 999 e -99999	999999999999999 999999999999999 999999	+0.999999999999999 999999999999999 998e-99939
12	Умножение на ноль	1e1	0	+0.0 E +0
13	Ноль на ноль	0	0	+0.0 E +0
14	Переполнение порядка	999999999999999 999999999999999 e 99999	999999999999999 999999999999999 999999	Переполнение порядка при выполнении умножения
15	Переполнение порядка	0.001 e -99999	1	Переполнение порядка при выполнении умножения
16	Ввод вещественного в обычной форме	10	10	+0.1 E +3
17	Начальные нули	00123e0	00123	+0.15129 E +5
18	Конечные нули	0.800 e 1	1	+0.8 E +1
19	Округление	4	999999999999999	+0.4 E +31

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

Самым выгодным по памяти вариантом является массив элементов типа char. Так же можно использовать целые типы (int, short int), однако по сравнению с массивом символов массив целых чисел будет занимать больше памяти.

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

Производить действия над числами поэлементно, заранее сохранив цифры числа в массив. Для выполнения операций использовать алгоритм столбиком или его модификации.

Вывод

В ходе выполнения лабораторной работы был успешно реализован алгоритм умножения длинного целого числа на длинное вещественное число, были получены навыки хранения чисел, не входящих в диапазон представления в ПК, и работы с ними.

Было выяснено, что для хранения чисел, выходящих за рамки машинного представления, эффективно использовать структуры для вещественных чисел – с полями, соответствующими знаку (символьный тип), мантиссе (массив символов), её длине (короткое целое), порядку (целое) и положению точки (короткое целое), для целых – самому числу со знаком (массив символов) и его длине (короткое целое).