



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 Егорова Полина Александровна

Группа ИУ7 – 34Б

Преподаватель Силантьева Александра Васильевна

2021 г.

ОПИСАНИЕ УСЛОВИЯ ЗАДАЧИ

Смоделировать операцию умножения двух действительных чисел в форме $+/-m.n E +/-K$, где суммарная длина мантиссы ($m+n$) - до 30 значащих цифр, а величина порядка K - до 5 цифр. Результат выдать в форме $+/-0.m1 E +/-K1$, где $m1$ - до 30 значащих цифр, а $K1$ - до 5 цифр.

ОПИСАНИЕ ТЕХНИЧЕСКОГО ЗАДАНИЯ

Входные данные:

2 действительных числа, где действительное число - строка, содержащая вещественное число в виде $<+/-m.nE+/-K>$. Знак перед числом и перед порядком обязательно. Знак экспоненты $<E>$ необязательно. Суммарная длина $<m+n>$ - до 31 цифры, включая точку; длина порядка — до 5 цифр.

Выходные данные:

Длинное число в виде $<+/-0.m1E+/-K1>$. Длина мантиссы $<m1>$ - до 30 цифр; длина порядка $<K1>$ — до 5 цифр.

Действие программы:

Умножение двух вещественных чисел.

Обращение к программе:

Запуск через терминал. (`./app.exe`)

Аварийные ситуации:

1. Некорректный ввод: строка с действительным числом содержит недопустимый символ (не цифра и не символ из набора («+», «-», «.», «E»)).

Сообщение на выходе: *«The entered string must not contain letters.»*

2. Некорректный ввод: переполнение порядка при вводе вещественного числа. Порядок превышает по модулю 99999.

Сообщение на выходе: «*Overflow power. Check the rules.*»

3. Некорректный ввод: превышение длины при вводе вещественного числа (больше 30 цифр, не включая точку и знак).

Сообщение на выходе: «*Incorrect input (length of number). Check the rules.*»

4. Некорректный ввод: отсутствие знака у мантиссы числа.

Сообщение на выходе: «*Put a number sign. Check the rules.*»

ОПИСАНИЕ СТРУКТУРЫ ДАННЫХ

После ввода числа, оно записывается в строку, после чего проходит обработку и далее хранится в массиве символов длиной 39 (с учетом всех служебных знаков: точка <.>, знак экспоненты <E>, знак числа и знак порядка <+>). Далее число обрабатывается и записывается в структуру `number_r`:

```
#define MAX_MANTISSA 32
#define MAX_POWER 8
#define MAX 59
#define MAX_LEN_NUM 39
```

```
char first_num[MAX_LEN_NUM];
char second_num[MAX_LEN_NUM];
```

```
typedef struct
{
    char sing_num;
    char mantissa[MAX_MANTISSA];
    char sing_power;
    char power[MAX_POWER];
    int add_power;
    int point;
    int exp;
    int int_mantissa[MAX_MANTISSA];
    int len_mantissa;
    int int_power;
} number_r;
```

Поля структуры:

`sing_num` – знак числа (+ или -).

`mantissa` – строка - мантисса числа.

`sing_power` – знак степени (+ или -).

`power` – строка - порядок вещественного числа (для целого его значение будет «+0»).

add_power – количество разрядов, на которое нужно домножить число, чтобы избавиться от точки.

point – позиция знака точки в мантиссе (если нет, то значение равно -41).

exp – позиция знака экспоненты (если нет, то значение равно -41).

int_mantissa – массив из чисел – символов мантииссы.

len_mantissa – длина мантииссы.

int_power – число - порядок вещественного числа.

ОПИСАНИЕ АЛГОРИТМА

1. Программа считывает две строки – два действительных числа.
2. Строки обрабатываются и записываются в структуру number_r, далее проводится проверка данных.
3. Если происходит переполнение мантииссы или порядка, либо порядок или мантиисса равна машинному нулю, то выводится поясняющее сообщение.
4. Если все данные верны, число приводится к нормализованному виду для упрощения дальнейших операций над них.
5. Происходит умножение первого числа на второе по методу «умножение в столбик» с контролем округления, если количество знаков результирующего числа превысило 30.
6. После деления результат выводится в нормализованном виде в соответствии со спецификацией, указанной в техническом задании (<+|-0.m1E+|-K1>).

НАБОР ТЕСТОВ

№	Название теста	Число №1	Число №2	Вывод
1.	Обычный тест	+89.547E+3	+1.1E-1	+0.985017E4
2.	Числа разных знаков	-4.4E-3	+9.45E2	-0.4158E1
3.	Отсутствие записи целой части	-.1	+0.4	-0.4E-1
4.	Умножение нуля на число	+0	+0.45	+0.0E0

19.	Превышение мантиисы	длины +123	+0.99..99 (30 девяток в дробной части, всего 31 знак)	Incorrect input. Check the rules.
20.	Превышение длины порядка	+15.6E1	+0.9E111111	Overflow power. Check the rules.
21.	Некорректный ввод	+a	+0.1	The entered string must not contain letters.
22.	Некорректный ввод	+0.1	+0.2Edkjf	The entered string must not contain letters.
23.	Некорректный ввод	+0.1EE3	+0.1	The entered string must not contain letters.
24.	Некорректный ввод	+ -0.1	+0.1	The entered string must not contain letters.
25.	Некорректный ввод	+123E+45	-	The entered string must not contain letters.
26.	Некорректный ввод	+0.4E-	+0.1	The entered string must not contain letters.
27.	Некорректный ввод	+12.3	+a	The entered string must not contain letters.
28.	Некорректный ввод	+5+10	+0.7	The entered string must not contain letters.
29.	Некорректный ввод	-0.7	-0.3-E-3	The entered string must not contain letters.

30.	Переполнение порядка	-0.001E-99999	-1	Overflow power. Check the rules.
31.	Переполнение порядка	-100.000E99999	-100	Overflow power. Check the rules.
32.	Отсутствие мантиссы	+E3	+0.1	The entered string must not contain letters.

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Каков возможный диапазон чисел, представляемых в ПК?

Диапазон чисел зависит от выбранного типа данных, разрядности процессора. Чем больше разрядность процессора, тем больше чисел он может вместить.

Максимальное значение беззаконного целого числа, для которого выделяется 64 разряда, равно $2^{64} - 1 = 18\,446\,744\,073\,709\,551\,615$.

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

Точность представления вещественных чисел определяется количеством памяти, выделяемой для хранения мантиссы числа. Для мантиссы числа типа float (4 байта) выделяется 24 бита, для double (8 байт) выделяется 52 бита, поэтому мантисса числа может иметь значение до 4 503 599 627 370 496 – 16 знаков.

3. Какие стандартные операции возможны над числами?

Возможны операции сложения, вычитания, умножения, деление, взятия остатка и сравнение.

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

Программист может выбрать структуру или массив символов. В структуру могут быть записаны знаки мантиссы и порядка, сами мантисса и порядок и любая иная вспомогательная информация. Аналогично можно работать и с массивом символов.

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

Необходимо создать собственные библиотеки, содержащие функции, обрабатывающие по отдельности каждый разряд числа, выходящего за рамки машинного кода.

ВЫВОД

Начиная выполнение какой-либо задачи, необходимо понимать, с данными какого типа и рода предстоит работать. Возможно, диапазон численных значений невелик, тогда для оптимизации работы программы и сохранения памяти можно использовать типы данных, диапазон значений которых мал. Возможна и обратная ситуация, когда диапазон предоставляемых данных выходит за рамки машинного представления. В таком случае необходимо позаботиться о корректной работе даже с очень большими числами и подготовить соответствующие библиотеки и функции.