



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 Дьяченко Артём Александрович

Группа ИУ7 – 33Б

Преподаватель Барышникова М. Ю.

Оглавление

<u>ОПИСАНИЕ УСЛОВИЯ ЗАДАЧИ.....</u>	<u>3</u>
<u>ОПИСАНИЕ ТЕХНИЧЕСКОГО ЗАДАНИЯ.....</u>	<u>3</u>
<u>ОПИСАНИЕ СТРУКТУРЫ ДАННЫХ.....</u>	<u>5</u>
<u>ОПИСАНИЕ АЛГОРИТМА.....</u>	<u>7</u>
<u>НАБОР ТЕСТОВ.....</u>	<u>8</u>
<u>ВЫВОД.....</u>	<u>9</u>

ОПИСАНИЕ УСЛОВИЯ ЗАДАЧИ

Смоделировать операцию умножения целого числа длиной до 30 десятичных цифр на действительное число 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:

Целое число: строка, содержащая целое число в виде $<+/-m>$. Если не указать знак перед числом – по умолчанию будет считаться за '+'. Длина модуля числа $<m>$ - до 30 цифр.

Действительное число: строка, содержащая вещественное число в виде $<(+/-)m.nE/(+/-)K>$. Если не указать знак перед числом и/или экспонентой – по умолчанию будет считаться за '+'. 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 целого числа на действительное.

Обращение к программе:

Запускается через терминал: вводится действительное число — за ним — целое.

Аварийные ситуации:

1. Некорректный ввод: превышение длины при вводе действительного числа (больше 39 цифр).
Код ошибки – 101.
2. Некорректный ввод: превышение длины при вводе целого числа (больше 31 цифры).
Код ошибки – 102.

3. Некорректный ввод: строка с действительным числом не подходит под регулярное выражение-маску `"^([+-]?([0-9]+([.][0-9]*)?([eE][+-]?[0-9]+)?|([.][0-9]+([eE][+-]?[0-9]+)?))$"`
Код ошибки – 103.
4. Некорректный ввод: строка с целым числом не подходит под регулярное выражение-маску `"^([+-]?[0-9]+)$"`
Код ошибки – 104.
5. Некорректный ввод/переполнение в вычислениях:
превышение длины порядка (больше 5 символов).
Код ошибки – 105.
6. Превышение длины мантиисы действительного числа.
Код ошибки – 106.
7. Превышение длины целого числа.
Код ошибки – 107.
8. Превышение длины действительного числа.
Код ошибки — 108.
9. Ввод пустой строки.
Код ошибки — 109.

Во всех аварийных случаях в консоль выводится сообщение:
"Невозможно провести расчёт! Введены некорректные данные!"
и/или сообщение, локализирующее ошибку.

ОПИСАНИЕ СТРУКТУРЫ ДАННЫХ

После ввода числа, оно хранится в массиве символов длиной 31 (для целого числа) или 39 (для действительного числа).

Далее действительное число обрабатывается и записывается в структуру `real_t`.

Структура `real_t`:

```
struct real_t
{
    int m_sign;
    int mantiss[MANTISS_SIZE];
    int exp_sign;
    int exp;
    int start_index;
};
```

Поля структуры:

`m_sign` – знак мантиссы

`mantiss` — мантисса

`exp_sign` — знак экспоненты (порядка)

`exp` — значение экспоненты (порядка)

`start_index` — индекс, с которого идёт вывод значений мантиссы; индекс последнего элемента мантиссы действительного числа.

Целое число обрабатывается и записывается в структуру `int_t`.

Структура `int_t`:

```
struct int_t
{
    int sign;
    int digits[MANTISS_SIZE];
    int start_index;
```

};

Поля структуры:

sign – знак числа

digits — цифры числа

start_index — индекс, с которого идёт вывод значений; индекс последней цифры целого числа.

Структура `res_t`, в которую помещается результат умножения, аналогична структуре `real_t`, за исключением длины мантиссы: она равняется `MANTISS_SIZE * 2`, т.е. 60.

ОПИСАНИЕ АЛГОРИТМА

1. Программа считывает две строки, одна содержит действительное число, другая — целое. Оба отличны от нуля.
2. Строки обрабатываются и записываются в структуры `real_t` и `int_t`.
3. Создаётся новая переменная структуры `res_t`, в которой сразу вычисляются знаки мантиссы и порядка, мантисса инициализируется нулями.
4. Происходит перемножение двух чисел с сохранением вычислений в переменную результата.
5. Если **справа** в мантиссе результирующей переменной есть нули – они убираются, переносятся в порядок.
6. Т.к. мантисса в выводе не может иметь больше 30 знаков, то происходит округление 30-й цифры мантиссы с переносом остатков.
7. Если случилось переполнение мантиссы, то все числа сдвигаются влево на одну позицию, а порядок увеличивается на единицу.
8. Если **слева** в мантиссе результирующей переменной есть нули – они убираются, переносятся в порядок.
9. Корректировка порядка экспоненты — она увеличивается на значение `RES_MANTISS_SIZE - res_num.start_index`, т. к. именно столько цифр нужно сдвинуть в дробную часть, после нуля в выводе.
10. Перевод данных результирующей переменной структуры `res_t` в строку и её вывод в виде `<(+|-)0.mE(+|-)K>`

НАБОР ТЕСТОВ

[illegible]

14	Некорректный ввод	100+E1	1	Невозможно провести расчёт!
15	Некорректный ввод	1	1.0	Невозможно провести расчёт!
16	Некорректный ввод		23	Пустая строка!
17	Некорректный ввод	23		Пустая строка!
18	Некорректный ввод	-123	+123.45E+6. 7	Невозможно провести расчёт!
19	Некорректный ввод	E123	1	Невозможно провести расчёт!

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Каков возможный диапазон чисел, представляемых в ПК?

Диапазон чисел зависит от разрядности процессора и выбранного типа переменной. Максимальное значение 64-разрядного беззаконного целого числа равно 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 double выделяется 52 бита, с помощью этого мантисса числа может иметь значение до 4 503 599 627 370 496.

3. Какие стандартные операции возможны над числами?

Операции сложения, вычитания, умножения, деление, взятие остатка, сравнения, возведения в степень.

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

Мы можем написать структуру, где можно записать мантиссу, знак числа и порядка. Также может использовать массив символов.

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

Можно использовать самостоятельно разработанные функции или библиотеки. Те, которые поразрядно умножают/складывают числа.

Вывод

Идея длинной арифметики мне знакома давно, её реализовывал я ещё в школе, но на C++ с использованием динамических массивов. В этой работе мне было легко придумать общий концепт решения и декомпозировать задачу. Отладка заняла в 2 раза больше времени, чем написание кода. Т.к. в этот раз я писал на C со статическими массивами, поэтому мне приходилось пристально следить за границами массива. В целом, я повторил и закрепил знания о длинной арифметике и представлении числа.