|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **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 Нисуев Нису Феликсович

Группа ИУ7 – 32Б

Преподаватель Барышникова Марина Юрьевна

2023 г.

**Оглавление**

[**Оглавление** 2](#_Toc145350457)

[**Описание условия задачи** 3](#_Toc145350458)

[**Описание технического задания** 3](#_Toc145350459)

[**Описание структуры данных** 4](#_Toc145350460)

[**Описание алгоритма** 5](#_Toc145350461)

[**Набор тестов** 6](#_Toc145350462)

[**Ответы на контрольные вопросы** 6](#_Toc145350463)

[**Вывод** 8](#_Toc145350464)

# **Описание условия задачи**

Смоделировать операцию умножения целого числа длиной до 40 десятичных цифр на действительное число в форме (+/-)m.n Е (+/-)K, где суммарная длина мантиссы (m+n) - до 30 значащих цифр, а величина порядка K - до 5 цифр. Результат выдать в форме (+/-)0.m1 Е (+/-)K1, где m1 - до 40 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 будется считаться за ‘+’. Длина модуля числа <m> - до 40 цифр.

Действительное число:

строка, содержащая вещественное число в виде <(+\-)m.nE\e(+\-)K>. Если не указать знак перед числом и/или экспонентой– по умолчанию будется считаться за ‘+’. Суммарная длина <m+n> - до 30 цифр, включая точку; длина порядка — до 5 цифр.

## **Выходные данные:**

Длинное число в виде <(+\-)0.m1E(+\-)K1>. Длина мантиссы m1 - до 40 цифр; длина порядка K1 — до 5 цифр.

## **Действие программы:**

Умножение целого числа на действительное

## **Обращение к программе:**

Запуск через терминал (./build/app.exe). Сначала вводится целое число, потом вещественное.

## **Аварийные ситуации:**

1. Некорректный ввод: превышение длины при вводе целого числа (больше 40 значащих цифр).
2. Некорректный ввод: превышение длины при вводе действительного числа (больше 30 значащих цифр).
3. Некорректный ввод: В числах присутствуют недопустимые символы
4. Некорректный ввод/переполнение: превышение длинны порядка (больше 5 символов).
5. Некорректный результат: Порядок полученного числа превышает 5
6. Некорректный результат: Размер мантиссы полученного числа превышает 40

# **Описание структуры данных**

Целое число обрабатывается при вводе и записывается в структуру **lllong\_t**.

Структура **lllong\_t**:

|  |
| --- |
| **typedef** struct **{**  bool sign**;**  char digits**[**LLONG\_LEN **+** 1**];**  **}** lllong\_t**;** |

Поля структуры:

* sign – знак числа
* digits – массив цифр числа

Действительное число обрабатывается при вводе и записывается в структуру **lexp\_t**.

Структура **lexp\_t**:

|  |
| --- |
| **typedef** struct **{**  bool sign**;**  char mantiss**[**LEXP\_LEN **+** 1**];**  int num\_order**;**  **}** lexp\_t**;** |

Поля структуры:

* sign – знак числа
* mantiss – массив цифр мантиссы экспоненциального числа
* num\_order – порядок экспоненциального числа

Результирующее число вычисляется и записывается в структуру **llexp\_t**. Структура результирующего числа ничем не отлучается от стуктур **lexp\_t**, кроме размера мантиссы

Структура **llexp\_t**:

|  |
| --- |
| **typedef** struct **{**  bool sign**;**  char mantiss**[**LLEXP\_LEN **+** 1**];**  int num\_order**;**  **}** lexp\_t**;** |

Поля структуры:

* sign – знак числа;
* mantiss – массив цифр мантиссы экспоненциального числа;
* num\_order – порядок экспоненциального числа.

Размеры массивов чисел:

|  |
| --- |
| #define LLONG\_LEN 40 // Максимальный размер целого числа  #define LLEXP\_LEN 40 // Максимальный размер мантиссы длинного экспоненциального числа  #define LEXP\_LEN 30 // Максимальный размер мантиссы короткого экспоненциального числа  #define ORDER\_LEN 5 // Максимальный порядок экспоненциальных чисел |

# **Описание алгоритма**

1. Программа считывает строку целого числа
2. Если в строке не найдено ошибок, то программа записывает число в структуру lllong\_t, иначе сообщает об ошибке и завершает программу
3. Программа считывает строку действительного числа
4. Если в строке не найдено ошибок, то программа записывает число в структуру lexp\_t с определенными преобразованиями, иначе сообщает об ошибке и завершает программу
5. Если все данные корректны, происходит умножение первого числа на второе методом умножения в столбик
6. После умножения программа записывает получившиеся значение в результирующую структуру llexp\_t с округлением, если оно требуется
7. После умножения результат выводится в нормализованном виде в соответствии со спецификацией, указанной в техническом задании (<+\-0.m1E+\-K1>). При порядке превышающем допустимый в техническом задании выводится ошибка.

# **Набор тестов**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Название теста | Число №1 | Число №2 | Вывод |
| Позитивные тесты | | | | |
| 1 | Умножение на единицу | 1 | 123.456e789 | +0.123456e792 |
| 2 | Умножение на десять | 10 | 1E5 | +0.1e7 |
| 3 | Умножение чисел разных знаков | +13 | -13 | -0.169e3 |
| 4 | Число меньше единицы | 3 | 0.15 | +0.45e0 |
| 5 | Умножение целых чисел | 21 | +12345 | +0.259245е6 |
| 6 | Граничные значения  (действительное число) | 1 | 999...999  (30 девяток) | +0.999999999999999999999999999999е30 |
| 7 | Граничные значения  (целое число) | +999...999  (40 девяток) | 1 | +0.999999999999999999999999999999е40 |
| 8 | Произведение нулей | 0 | 0 | +0.0е0 |
| 9 | Произведение единиц | 1 | 1 | +0.1е1 |
| 10 | Произведение с округлением | 999…999 (40 девяток) | 2 | +0.2e41 |
| **Негативные тесты** | | | | |
| 1 | Превышение длины порядка | 2 | 123e10000 | ERROR: Incorrect range |
| 2 | Превышение длины мантиссы (действительного числа) | +1 | 999...999  (31 девятка) | ERROR : Incorrect input |
| 3 | Некорректный ввод | 123 | abc | ERROR : Incorrect input |
| 4 | Некорректный ввод | abc | Не вводится | ERROR : Incorrect input |
| 5 | Некорректный ввод | 1 | 23a | ERROR : Incorrect input |
| 6 | Некорректный ввод | 23a | Не вводится | ERROR : Incorrect input |
| 7 | Некорректный ввод | 1 | 100+E1 | ERROR : Incorrect input |
| 8 | Некорректный ввод | 1.0 | Не вводится | ERROR : Incorrect input |
| 9 | Некорректный ввод | 23 | --23 | ERROR : Incorrect input |
| 10 | Некорректный ввод | --23 | Не вводится | ERROR : Incorrect input |
| 11 | Некорректный ввод | +123.45E+6.7 | Не вводится | ERROR : Incorrect input |
| 12 | Некорректный ввод | 1 | E123 | ERROR : Incorrect input |
| 13 | Слишком маленькое получившееся число | 1 | 0.0001e-99999 | ERROR: Calculated num out of acceptable values |
| 14 | Слишком большое получившееся число | 100 | 1e99999 | ERROR: Calculated num out of acceptable values |

# **Ответы на контрольные вопросы**

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

Диапазон доступных чисел зависит от типа данных, который выбран, и разрядности процессора. Чем выше разрядность процессора, тем больший диапазон чисел он способен обрабатывать. Например, если процессор имеет 64 разряда, то максимальное значение для беззнакового целого числа составляет 18 446 744 073 709 551 615, что эквивалентно 2^64 - 1.

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

Точность представления вещественных чисел зависит от объема памяти, отведенного для хранения мантиссы числа. Например, для типа данных float (4 байта) выделяется 24 бита для мантиссы, а для double (8 байт) - 52 бита. Это позволяет мантиссе числа иметь значение до 4 503 599 627 370 496 – 16 знаков.

*3. Какие стандартные операции возможны над числами?*

Над числами возможно выполнять следующие 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 определить выбор между использованием структуры или массива символов. Структура может содержать информацию о знаках мантиссы, порядке, мантиссе и порядке, а также другие дополнительные данные. Точно так же, можно оперировать с массивом символов.

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

Требуется разработать собственные библиотеки, которые будут включать в себя функции для обработки отдельных разрядов чисел, которые выходят за пределы машинного кода.

# **Вывод**

В ходе этой работы я научился обрабатывать и работать с числами выходящими за рамки в используемом языке. Основная задача для программиста правильно выбрать структуру данных и реализовать операции над числами выходящими за рамки числовой сетки.