|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство образования и науки 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 (ИУ)

КАФЕДРА Программное обеспечение ЭВМ и информационные технологии (ИУ7)

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1  
«ОБРАБОТКА БОЛЬШИХ ЧИСЕЛ»**

Студент, группа **Буланый К., ИУ7-36Б**

*2020 г.*

**Описание условия задачи**

Смоделировать операцию умножения действительного числа длиной̆ до 30 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 30 значащих цифр, а K1 - до 5 цифр.

**Техническое задание**

**Входные данные:**

1. **Действительное число:** строка, содержащая вещественное число в виде

[+/-]m[.n][e[+/-]K]  
Суммарная длина мантиссы (m+n) – до 30 цифры, порядка K – до 5 цифр. Наличие точки, знака порядка и знака числа обязательно. Наличие знака экспоненты e/E необязательно.

1. **Действительное число:** строка, содержащая вещественное число в виде

[+/-]m[.n][e[+/-]K]  
Суммарная длина мантиссы (m+n) – до 30 цифр, порядка K – до 5 цифр. Наличие точки, знака порядка и знака числа обязательно. Наличие знака экспоненты e/E необязательно.

**Выходные данные:**

* длинное число, нормализованное в виде [-]0.m1 e[-]K1, где длина m1 – до 30 цифр, K1 – до 5.

**Функция программы:** умножение действительного числа на действительное.

**Обращение к программе:** запускается из терминала.

**Аварийные ситуации:**

1. Некорректный ввод строки с вещественным числом.  
   На входе: строка, хотя бы один символ в которой не цифра и не символ из набора “+ - . e E”.  
   На выходе: сообщение «Invalid characters input. Try again.».
2. Превышение длины строки при вводе мантиссы.  
   На входе: корректное число, длина мантиссы которого превышает 30 цифр.  
   На выходе: сообщение «Invalid characters in mantissa OR it is too long. Try again.»
3. Превышение длины строки при вводе порядка.  
   На входе: корректное число, длина порядка превышает 5 цифр.  
   На выходе: сообщение «Invalid characters in exponent OR it is too long. Try again.»
4. Переполнение порядка.  
   На входе: в процессе умножения степень, полученного в результате числа превышает 99999.  
   На выходе: сообщение «Exponent oveflow while multiplying.»

**Допущения**

Если введена очень длинная строка, фрагмент которой длиной **LEN** будет корректен, то строка будет принята и сообщение об ошибке не будет выведено, так как считывается сразу вся строка. При этом незначащие нули пропускаются.

**Структуры данных**

Для изначального хранения введенного числа используется символьный массив **str**. Он описывается как:

**char str[LEN],**

где **LEN –** максимальная длина массива с учетом всех знаков.

После ввода числа, оно обрабатывается и разбивается на части, которые описываются в структуре **number\_t**. Он описывается как:

**typedef struct number**

**{**

**int sign\_m;**

**int mantissa[MAX\_MANTISSA];**

**size\_t len\_m;**

**int exponent;**

**} number\_t;**

Поля структуры:

* **sign\_m** – знак числа. Принимает значения “-1” или “1”;
* **mantissa[MAX\_MANTISSA]** – значение мантиссы числа (часть между знаком числа и знаком экспоненты), **MAX\_MANTISSA** = 30;
* **len\_m** – длина мантиссы;
* **exponent** – экспонента.

**Алгоритм**

1. На вход программе подаётся первая строка символов, содержащая вещественные числа.
2. Проверяется корректность введенной строки.
3. Выполняется запись полученного числа в структуру типа **number\_t**.
4. На вход программе подаётся вторая строка символов, содержащая вещественные числа.
5. Проверяется корректность введенной строки.
6. Выполняется запись полученного числа в структуру типа **number\_t**.
7. Выполняется умножение чисел методом “умножения в столбик”, при этом контролируется округление.
8. При переполнении порядка пользователю выводится соответствующее сообщение об ошибке.
9. При некорректном вводе выводится соответствующее сообщение.
10. Если результат умножения корректен, то он выводится пользователю в нормализованном виде в соответствии со спецификацией, указанной в ТЗ.

**Тесты**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Тест | Число 1 | Число 2 | Результат |
| 1 | Некорректный̆ ввод | qwe | - | Invalid characters input. Try again. |
| 3 | Некорректный̆ ввод | 123 | +1.23.45 | Invalid characters in mantissa OR it is too long. Try again. |
| 4 | Некорректный̆ ввод | 123e45.67 | - | Invalid characters in mantissa OR it is too long. Try again. |
| 5 | Некорректный̆ ввод | 123.2 | e123 | Invalid characters in mantissa OR it is too long. Try again. |
| 6 | Превышение длины мантиссы | 999...999 (31 девятка) | - | Invalid characters in mantissa OR it is too long. Try again. |
| 7 | Превышение длины порядка | 123 | 0.1e123456 | Invalid characters in exponent OR it is too long. Try again. |
| 8 | Округление | 99…99 | 0.5 | 0.5e30 |
| 9 | Переполнение порядка | 1e-99999 | 1e-10 | Exponent oveflow while multiplying. |
| 10 | Умножение целых чисел | 100 | 5. | 0.5e3 |
| 11 | Умножение чисел разных знаков | 5 | -1e5 | -0.5e-4 |
| 12 | Граничные значение | 99…9e99999 (30 девяток) | 99…99e-99999 (30 девяток) | 0.999999999999999999999999999998e60 |
| 13 | Умножение на ноль | 0 | 0 | 0.0e0 |

**Контрольные вопросы**

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

Возможный диапазон чисел зависит от их типа, размера выделенной для их хранения памяти, разрядности процессора. Для беззнакового целого числа выделяется 64 двоичных разряда.

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

Точность представления вещественных чисел зависит от количества памяти, выделенного для хранения мантиссы. Для хранения мантиссы числа двойной точности (double) выделяется 52 двоичных разряда.

**3. Какие стандартные операции возможны над числами?**

Предусмотрено выполнение сравнения, сложения, вычитания, умножения, деления, взятия остатка от деления.

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

Программист может выбрать массив цифр для представления длинных чисел, а также структуру, в которую можно включить не только мантиссу числа, но и порядок со знаком.

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

Для этого можно использовать специальные библиотеки, а также самостоятельно разработанные решения.

**Вывод**

При выполнении лабораторной работы была реализована обработка и операция умножения длинных чисел. При работе с длинной арифметикой эффективно использовать массив цифр числа.