



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 ПРОГРАММНАЯ ИНЖЕНЕРИЯ (ИУ7)

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.03.04** Программная инженерия

О Т Ч Е Т

По лабораторной работе № 1

Название: Обработка больших чисел

Дисциплина: Типы и структуры данных

Студент

ИУ7-31Б

(Группа)

К. В. Морозов

(И.О. Фамилия)

Преподаватель

Москва, 2022

Условие задачи

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

Описание ТЗ

1. Описание входных данных

На вход поступает 2 строки:

- 1-я вида: $(+ | -)XX...X$, где:
 - X – десятичная цифра ($0...9$)
 - $(+ | -)$ – возможный знак плюс или минус
- 2-я вида: $(+ | -)XX...X(.)XX...X(e | E)(+ | -)XX...X$, где:
 - X – десятичная цифра ($0...9$)
 - $(+ | -)$ – возможный знак плюс или минус
 - $(.)$ – возможная точка
 - $(e | E)$ – возможный вид экспоненциальной формы записи

Ограничения и особенности реализации:

- Максимальная разрешенная длина целого числа в первом поле – 30 десятичных цифр
- Максимальная длина мантииссы во втором поле – 30 значащих цифр
- Максимальная длина порядка во втором поле – 5 десятичных цифр
- В оба поля ввода запрещается вводить неликвидные символы
- Ограничение длины первой строки – 31 символ
- Ограничение длины второй строки – 40 символов
- Если явно не задан знак числа – число является положительным
- В порядке могут быть записаны первые незначащие нули
- Нельзя начинать ввод вещественной части с экспоненциальной формы
- Если используется экспоненциальная форма, то обязательно должна присутствовать хотя бы одна цифра после нее
- При вводе вещественного числа нельзя использовать экспоненциальную форму с нулём ($0(e | E)...$)
- Если число в первом поле не является нулём, то число не может с него начинаться
- Если число во втором поле не является нулём, то число не может с него начинаться при условии, что после него не стоит точка

- Если в первом и во втором поле введены нули, то они обязательно должны быть без знака

Примеры ввода первого числа:

Допустимый ввод	Недопустимый ввод
0	-0
123	/* пустой ввод */
+123	123.123
-123	++2
123456789012345678901234567890	1234567890123456789012345678901234
9123	A
-99999999999999999999999999999999	01234

Примеры ввода второго числа:

Допустимый ввод	Недопустимый ввод
123	123.123.123
123.123	Кошка
123.123e123	1234g123
123.123E-123	/* пустой ввод */
0.0	0e14
-14e-88888	1234567890123456789012345678901234
+88E+88	1e5000005
24E40	E40
+0.1234	+00.1234

2. Описание результата работы программы

Программа осуществляет ввод чисел в указанном диапазоне значений и выдаёт результат в нормализованной форме $\pm 0.m1 E \pm K1$, где число $m1$ определено до 30 значащих цифр, число $K1$ – до 5 цифр. При невозможности произвести вычисления выдаётся соответствующее сообщение. Если в процессе вычислений количество значащих цифр мантиисы превосходит 30, то происходит округление.

3. Описание задачи, реализуемой программой

Программа реализует умножение длинного целого числа (длиной до 30 десятичных цифр) и длинного вещественного числа (длина мантиисы до 30 значащих цифр, длина порядка по модулю не превосходит 5 десятичных цифр). Основной алгоритм, применённый в программе, по сути, выполняет умножение “столбиком” над значащими цифрами мантиисы.

4. Описание обращения к программе

Обращение к программе происходит посредством вызова исполняемого файла `app.out` из командной строки.

5. Описание возможных аварийных ситуаций и ошибок пользователя

- Некорректный ввод строк на вход программе
- Результат работы программы выходит за допустимые пределы (напр. более 5 цифр порядка)

Описание внутренних структур данных и алгоритма

1. Описание внутренних структур данных

Описание структуры, представляющей хранение числа в системе:

```
struct big_double
{
    char sign;
    int order;
    int mantissa[MAX_MANT_LEN];
    size_t n_sign_value;
};
```

- `char sign` – поле хранит знак числа
- `int order` – поле хранит значение порядка числа
- `int mantissa[MAX_MANT_LEN]` – поле хранит значение мантиссы числа
- `size_t n_sign_value` – поле хранит количество значащих цифр мантиссы значащего числа

2. Алгоритм

1. Начало
2. Ввод первого целого числа
- (2.) Вывод ошибки, при некорректности ввода
3. Ввод второго вещественного числа
- (3.) Вывод ошибки, при некорректности ввода
4. Парсирование строчного представления целого числа в структурное
5. Парсирование строчного представления вещественного числа в структурное
6. Умножение чисел
- (6.) Вывод ошибки, при выходе результата умножения за допустимые пределы
7. Вывод результата умножения в нормализованном виде

Тестирование программы

1. Позитивные тесты

- 1) Целое число – ноль (вещественное число - корректно)
- 2) Целое число – одна цифра (вещественное число - корректно)
- 3) Целое число – 30 цифр (вещественное число - корректно)
- 4) Целое число записано со знаком + (вещественное число - корректно)
- 5) Целое число записано со знаком – (вещественное число - корректно)
- 6) Вещественное число – ноль (целое число - корректно)
- 7) Вещественное число представлено как целое (целое число - корректно)
- 8) Вещественное число представлено как с целой частью, так и вещественной (целое число - корректно)
- 9) Вещественное число представлено как с целой частью, так и вещественной, так и экспоненциальной формой (целое число - корректно)
- 10) Вещественное число записано со знаком + (целое число - корректно)
- 11) Вещественно число записано со знаком – (целое число - корректно)
- 12) Вещественное число имеет 30 значащих цифр мантииссы (целое число - корректно)
- 13) Вещественно число имеет 5 цифр порядка (целое число - корректно)
- 14) Порядок вещественного числа записан с + (целое число - корректно)
- 15) Порядок вещественного числа записан с - (целое число - корректно)
- 16) Первое число положительно, второе отрицательно
- 17) Первое число отрицательно, второе положительно
- 18) Первое число отрицательно, второе число отрицательно
- 19) Результат умножения – округляется без изменения (число следующее за последним, которое нужно вывести ≤ 4)
- 20) Результат умножения – округляется с изменениями (число следующее за последним, которое нужно вывести ≥ 5)

[illegible]

2	
2 2.2	0.44e1
2 2.2e2	0.44e3
1 +1.1	0.11e1
1 -1.1	-0.11e1
2 0.123456789012345678901234567891	0.246913578024691357802469135782e0
3 3e55555	0.9e55556
3 3e+3	0.9e4
3 3e-3	0.9e-2
3 -3	-0.9e1
-3 3	-0.9e1
-3 -3	0.9e1
90000000000000000000000000000002 2	0.18e31
90000000000000000000000000000009 2	0.18000000000000000000000000000002e31

2. Негативные тесты

- 1) В поле целого числа введён пустой ввод
- 2) В поле целого числа введён длинный ввод
- 3) В поле целого числа введено много десятичных цифр
- 4) В поле целого числа введён некорректный символ
- 5) В поле вещественного числа введён пустой ввод
- 6) В поле вещественного числа введён длинный ввод
- 7) В поле вещественного числа введён некорректный символ
- 8) В поле вещественного числа введён слишком длинный по модулю порядок
- 9) В поле вещественного числа введена слишком длинная мантисса
- 10) В ответе получился слишком длинный по модулю порядок

Входные данные	Результат
/* пустой ввод */	Ошибка! Пустой ввод
99999999999999999999999999999999	Ошибка! Длинный ввод

[illegible]

Выводы

В ходе выполнения данной лабораторной работы были получены навыки работы с “длинными” числами, превышающими разрядную сетку персонального компьютера. Разработчик должен самостоятельно придумывать и реализовывать алгоритмы и структуры данных, которые будут позволять выполнение операций с такими числами.

Отвѣты на контрольные вопросы

- 1) Диапазон чисел в компьютере может изменяться в зависимости от следующих параметров: Размер выделенной под данное число области памяти, является ли данное число знаковым или беззнаковым, является ли данное число представленным в виде целого или вещественного.
- 2) Вещественные числа обычно хранятся и используются в представлении с плавающей точкой в виде: $X = M * E^{*p}$, где M – мантисса со знаком, E – основание (10 или 16), p – целый порядок со знаком. Если десятичная точка расположена в мантиссе перед первой значащей цифрой числа, то при фиксированном количестве разрядов, отведённых под мантиссу, 2 обеспечивается возможность сохранить максимальное количество значащих цифр, то есть обеспечить максимальную точность представления числа в ПК.
- 3) Сложение, вычитание, умножение, деление и сравнение целых чисел.
- 4) Программист может выбрать разные способы хранения и представления таких чисел. В общем случае – это может быть структура со следующими полями (количество и значение полей может изменяться в зависимости от выбранной реализации): знак числа (символьный тип), порядок числа (целочисленный тип), мантисса числа (целочисленный массив), количество значащих цифр числа (целочисленный тип).

- 5) Операции над такими числами удобно выполнять путём последовательного почленного выполнения операций над всеми цифрами, хранящимися в поле структуры, отвечающим за мантиссу.