**Условие задачи**

Составить программу умножения или деления двух чисел, где порядок имеет до 5 знаков: от –99999 до +99999, а мантисса – до 30 знаков.

**Техническое задание**

Смоделировать операцию деления вещественного числа в форме ±M.Ne±K, где суммарная длина мантиссы (M+N) - до 30 значащих цифр, а величина порядка K - до 5 цифр, на целое число, длиной до 30 значащих цифр. Результат выдать в форме ±0.MN±K1, где MN - до 30 значащих цифр, а K1 - до 5 цифр.

**Входные данные:**

1. Строка, в которой записано вещественное число (делимое) в одном из следующих форматов:

* ±M.Ne±K, где длина мантиссы (M+N) - до 30 значащих цифр, а величина порядка K - до 5 цифр;
* ±M, где длина мантиссы (M) - до 30 значащих цифр;
* ±M.N, где длина мантиссы (M+N) - до 30 значащих цифр.

2. Строка, в которой записано целое число (делитель) в следующем формате:

* ±M, где длина числа (M), не больше 30 значащих цифр.

Если вводимое число положительно, ввод ‘+’ опционален.

**Выходные данные:**

Строка, в которое записано частное от деления в следующем виде:

* ±0.MN±K1, где MN - до 30 значащих цифр, а K1 - до 5 цифр.

Либо, сообщение об ошибке.

**Аварийные ситуации:**

1. Некорректный ввод данных пользователем.
2. Переполнение порядка частного в ходе деления
3. Деление на ноль

**Структуры данных**

В программе используется только одна структура данных – это запись (структура) с именем *num\_t:*

typedef struct {

char sign; // знак мантисы

char mantissa[MAX\_MANTISSA + 2]; // мантисса числа, MAX\_MANTISSA = 30

int exponent; // порядок числа

} num\_t;

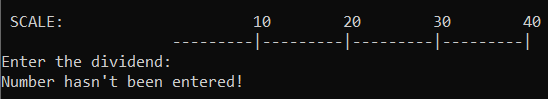
**Описание алгоритма**

В программе использован алгоритм “деления столбиком”. Мантиссы делимого и делителя приводятся к нормализованному виду. На очередной итерации делитель вычитается из первых разрядов мантиссы, пока остаток не будет меньше самого делителя; далее сносится очередная цифра из мантиссы делимого (сдвиг мантиссы влево на 1 позицию) – таким образом вычисляется очередная цифра частного. Стоит отметить, что никаких дополнительных структур данных (целочисленных массивов и т.п.) в алгоритме не используется. Кроме того, алгоритм предусматривает изменение порядка делимого вследствие нормализации его мантиссы и в случаях, когда деление нацело невозможно. Частное от деления выводится в нормализованном видe и при необходимости округляется. Если в ходе деления произошло переполнения порядка частного выдается сообщение о соответствующей ошибке.

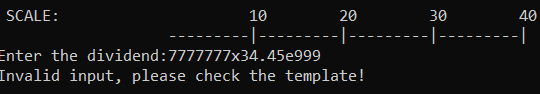
**Тесты**

**Аварийные тесты:**

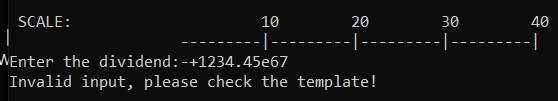
* Пустой ввод



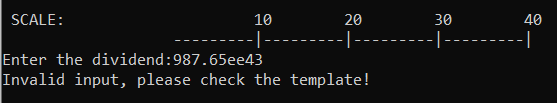
* Недопустимые символы

****

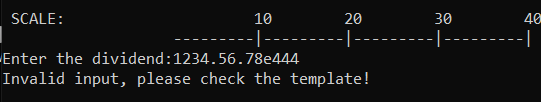
* Введены лишние знаки мантиссы

****

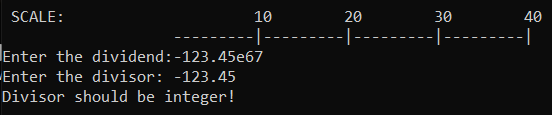
* Введено более одного знака порядка (‘e’) в делимом

****

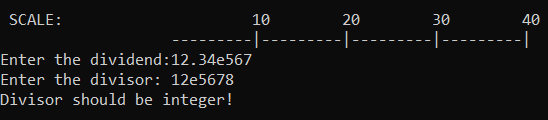
* В делимом присутствует более одной точки (‘.’)

****

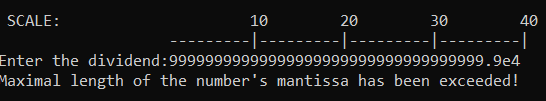
* Вещественный делитель

****

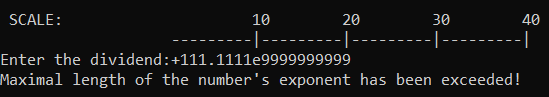
* Делитель записан в экспоненциальном виде

****

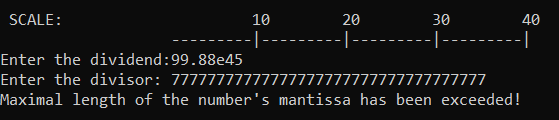
* Мантисса делимого содержит более 30 цифр

****

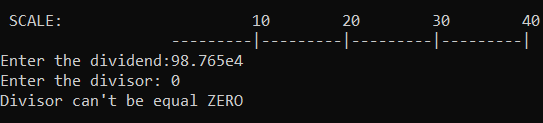
* Порядок делимого содержит более 5 цифр

****

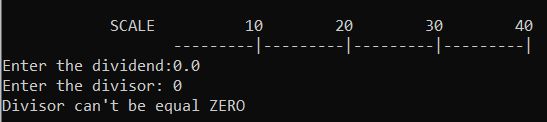
* Делитель содержит более 30 цифр

****

* Делитель равен нулю (деление на ноль)

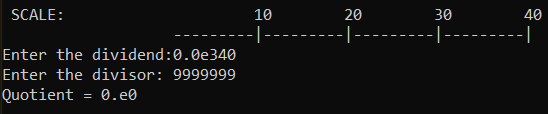
****

* И делитель, и делимое равны нулю

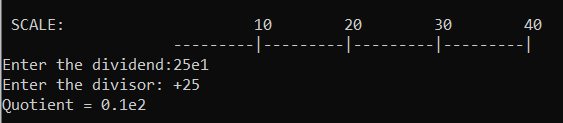
****

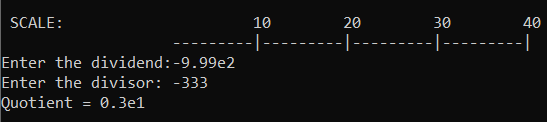
**Граничные значения:**

* Ноль в делимом

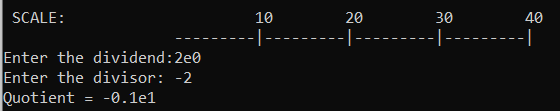


* Деление чисел с одинаковыми знаками (’+’/’+’ или ‘-‘/’-‘)

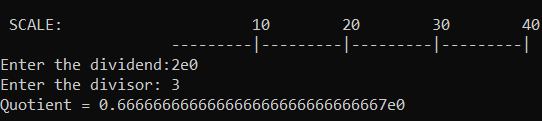




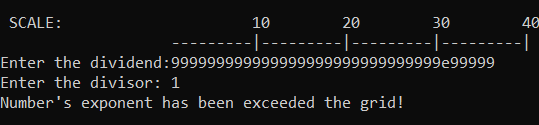
* Деление чисел с разными знаков (‘+’/‘-‘)



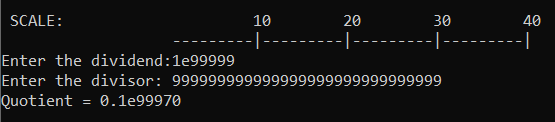
* Округление частного



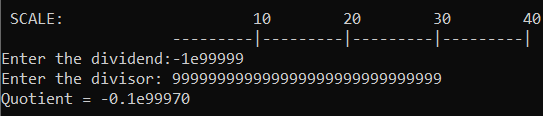
* Переполнение порядка частного



* Пограничное верхнее значение



* Пограничное нижнее значение



**Выводы по проделанной работе**

Для реализации операции деления над числами, выходящими за разрядную сетку ПК, лучше всего подходит алгоритм “столбиком” благодаря своей довольно простой реализации. Для работы с такими числами удобно использовать такую структуру данных, как запись (struct) с полями, содержащими данное число в следующей конструкции: знак мантиссы (char), мантисса (char[]), порядок (int).

**Контрольные вопросы**

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

Для n-разрядного машинного слова:

* целые положительные числа: 0 < X <= 2^n – 1
* целые отрицательные числа: -2^(n-1) <= X < 0

Числа с плавающей точкой*:* 3.6E–4951 <= x <= 1.1E+4932 (максимальный размер мантиссы 52 разряда, порядок – 11 разрядов)

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

Длина мантиссы определяется размером разрядной сетки. Максимально для представления мантиссы в памяти компьютера отводится 52 двоичных разряда.

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

Сложение, вычитание, умножение, деление, сравнение.

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

Можно хранить в целочисленном (символьном) массиве. Для этого у введенное числа можно поразрядно брать цифру и вставлять ее в отдельный элемент массива (элементы массива упорядочены в точности, как в исходном числе).

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

Операции над числами, выходящими за рамки машинного представления, можно реализовать, имплементируя алгоритмы сложения/деления/умножения/деления “столбиком” над структурами, в которых эти числа хранятся.