

Условие задачи:

Смоделировать операцию деления действительного числа на действительное число в форме $\pm m.nE\pm K$, где суммарная длина мантиссы ($m+n$) - до 30 значащих цифр, а величина порядка K - до 5 цифр. Результат выдать в форме $\pm 0.m1E \pm K1$, где $m1$ - до 30 значащих цифр, а $K1$ - до 5 цифр.

Техническое задание:

Исходные данные:

- передаются в строке формата $\pm m.nE\pm K$, допускается отсутствие знака числа, точки, мантиссы и порядка. Знак порядка обязателен, регистр $E(e)$ неважен.
- ($m+n$) - до 30 цифр, K - до 5 цифр.

Результат:

- передается в формате $\pm 0.m1E\pm K1$.
- $m1$ – до 30 цифр, $K1$ – до 5 цифр.

Задача:

Деление вещественного числа на вещественное, при выходе за 30 значащих цифр производится округление (если последующая ≥ 5 производится увеличение предыдущей цифры на 1, в противном случае цифра не меняется).

Аварийные ситуации:

- Некорректный ввод чисел (превышение допустимого количества, использование неразрешенных символов или их излишнее использование). Вывод сообщения на экран.
- Деление на 0. Вывод сообщения на экран.
- Переполнение порядка или машинный 0. Вывод сообщения на экран

Структуры данных:

Для хранения длинных чисел используется целочисленные массивы с максимальной длиной 30.

Порядок хранится в целочисленном типе.

Алгоритм:

1. Проверка корректности ввода.

Количество цифр числа не превышает допустимого (30) и для порядка (5), иначе код ошибки INCORRECT_INPUT 1.

Строка не содержит недопустимых знаков или количество допустимых знаков больше положенного (несколько точек или е, знаки плюса и минуса в некорректных местах или отсутствуют перед порядком). Код ошибки INCORRECT_INPUT 1.

Вначале как может быть знак, так и нет. При отсутствии цифр (но при наличии точки или знака) число рассматривается как ноль. После знака е цифр также может не быть, порядок также приравнивается к 0. Если цифры есть, перед ними должен стоять знак порядка. Символы кроме (+ - е) запрещены.

При выполнении всех условий возвращается SUCCESS 0.

2. Преобразование чисел.

Числа приводятся к виду $\pm A E \pm K$. Где А – само число, хранится посимвольно в массиве, К – порядок, хранится в переменной целочисленного типа. То есть все цифры, которые находятся после точки переносятся в целую часть, с уменьшением порядка числа.

На этом этапе убираются нули, это происходит также с изменением порядка.

Знак числа хранится в отдельной переменной, если знак отрицательный переменной присваивается значение 1, в другом случае 0.

При успешном выполнении возвращается SUCCESS 0.

3. Деление.

Проверка делимости на ноль. Код ошибки DIVISION_INTO_ZERO 2 и вывод сообщения на экран.

1) Производится дописывание нулей в конец у делимого до достижения максимального размера.

2) Выравнивается размер делителя с делимым, дописываем в конец делителя нули.

- 3) Вычитаем, пока возможно, из делимого делитель, считаем количество вычитаний. Записываем полученную цифру в массив результата.
- 4) Если вычитание прошло успешно (выполнилось хотя бы один раз) уменьшаем количество элементов у делимого и делителя. У делителя убираем последние нули до выравнивания с делимым. Делимое после вычитания имеет другое значение и соответственно количество цифр.
- 5) Пока делимое больше 0 выполняем пункты 3 и 4.

Пример:

123

1

_123

100 n = 1 – вычли один раз

_23

10

_13

10 n = 2 – вычли два раза

_3

1

_2

1

_1

1 n = 3 – вычли три раза

0

Вычитание закончено, делимое равно 0. Результат 123.

Добавление нулей в конец делимого вначале происходит для увеличения точности вычислений.

Проверка переполнения или машинного нуля. Код ошибок DIVISION_IMPOSSIBLE 3 и вывод сообщения на экран.

При успешном делении SUCCESS 0.

4. Вывод результата.

Частное приводится к нормализованному виду. Убираются нули в начале и в конце, за счет изменения порядка.

Итог вывод:

$\pm 0.m1e\pm K1$

Для примера:

$+0.123e+3$

Тесты:

1. Проверки на правильность ввода и вывода

Пустой ввод

```
C:\msys64\mingw64
```

```
|  
Input number  
Incorrect input_
```

Больше 30 цифр в числе

```


C:\msys64\mingw64\bin\c
Input number
1234567812345678912345678901234567890
Incorrect input_

```

Больше 5 цифр в порядке

```
C:\msys64\mingw64
Input number
234e-123456
Incorrect input_
```

Недопустимые символы



```

C:\msys64\mingw64
|
Input number
22425d45
Incorrect input_

```

2. Граничные значения

```
C:\msys64\mingw64\bin\gcc.exe -fcommon -c test.c -o test.o  
C:\msys64\mingw64\bin\gcc.exe -fcommon -c test.c -o test.o  
Input number  
9999999999999999999999999999999e+99999  
Input number  
9999999999999999999999999999999e-99999  
OVERFLOW_  
  
C:\msys64\mingw64\bin\gcc.exe -fcommon -c test.c -o test.o  
C:\msys64\mingw64\bin\gcc.exe -fcommon -c test.c -o test.o  
Input number  
9999999999999999999999999999999e+99999  
Input number  
1  
OVERFLOW
```


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?

Массив из цифр этого числа и его порядка, и переменные хранящие знаки числа и порядка.

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

Последовательно выполнять какие-либо операции с каждой цифрой числа, записанного в массив.

Вывод:

Числа имеют конечную длину, для длинных чисел, которые не входят в диапазон требуется использовать массивы для их хранения.

Выполнение операций производится над каждой цифрой с возможностью изменения порядка. Возможно возникновение переполнения. В итоге числа приводятся к нормализованному виду.