

Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный технический университет имени Н.Э. Баумана (напиональный исследовательский университет)»

(национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ

« Информатика и системы управления»

КАФЕДРА

«Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ №1 «ДЛИННАЯ АРИФМЕТИКА»

Группа ИУ7-33Б

Название предприятия НУК ИУ МГТУ имени Н. Э. Баумана

Студент Лысцев Никита Дмитриевич

Преподаватель Барышникова М. Ю.

2022 г.

Оглавление

1.	Описание условия задачи	. 2
	Описание технического задания	
	Описание внутренних структур данных	
4.	Описание алгоритма	. 5
5.	Набор тестов	. 5
6.	Выводы	.7
7.	Ответы на контрольные вопросы	.7

1. Описание условия 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.

2. Описание технического задания

2.1. Входные данные

Две строки, содержащие действительные числа в экспоненциальной или обычной форме.

Действительное число -- строка вида <±m.nE±K>, >, где суммарная длина мантиссы (m+n) - до 30 значащих цифр, а величина порядка K - до 5 цифр.

2.2. Выходные данные

Строка, содержащая в себе результат деления двух введенных действительных чисел.

Результат деления двух действительных чисел представляется в нормализованном виде, то есть в виде $<\pm 0.m1E\pm K1>$, где m1- результирующая мантисса (до 30 значащих цифр), а K1- результирующий порядок (до 5 цифр).

2.3. Описание задачи, реализуемой программой

Задача программы – осуществить деление одного действительного числа на другое действительное число.

2.4. Способ обращения к программе

Ввод и вывод всех данных осуществляется через консоль.

2.5. Описание возможных аварийных ситуаций и ошибок пользователя

Аварийные ситуации:

- Деление на ноль;
- Пустое поле ввода ничего не произойдет, программа будет ждать ввода пользователя;
- Переполнение порядка при делении.

Ошибки пользователя:

• Некорректный ввод: превышение допустимой длины мантиссы, превышение допустимого порядка, ввод не валидных данных.

3. Описание внутренних структур данных

Введенное число сохраняется в массиве символов, длиной 39 элементов.

Далее идет обработка полученных данных и заполнение полей структуры real_number_t.

```
typedef struct
{
   int mantissa[MAX_LEN_MANTISSA];
   int len_mantissa;
   char sing_num;
   int power;
   int point_pos;
   int e_pos;
} real_number_t;
```

Листинг 1. Структура real_number_t, содержащая в себе составные части введенного числа

- 1. mantissa массив целых чисел, элементами которого являются цифры мантиссы введенного числа.
- 2. len_mantissa количество элементов в мантиссе.
- 3. sing_num знак числа.
- 4. power целое число, порядок введенного числа.

- 5. point_pos позиция точки в мантиссе.
- 6. e_pos позиция экспоненты в числе.

4. Описание алгоритма

Программа предлагает пользователю ввести два действительных числа, считывает введенные строки и проверяет каждую из них на валидность. Если полученные данные некорректны, то выводится соответствующее сообщение.

По алгоритму деления в столбик последовательно вычисляются цифры и порядок результирующего числа. Шаг деления происходит следующим образом: оба числа нормализуются, запоминается разность их порядков, порядки отбрасываются. Если знаменатель оказывается больше числителя, он смещается вправо на один разряд. Пока числитель больше знаменателя, производится вычитание знаменателя из числителя с инкрементом значения счетчика. Значение счетчика после выполнения цикла — очередная десятичная цифра результирующего числа. Сохраненная разность порядков — смещение относительно предыдущих цифр.

Результат деления двух чисел выводится в виде $<\pm 0.m1E\pm K1>$.

5. Набор тестов

№	Название	Число №1	Число №2	Результат
1	Некорректны	bbb	-	ERROR: a real number is
	й ввод			entered in an invalid
				form!
2	Некорректны	+ddds.dfdsfE+54	-	ERROR: a real number is
	й ввод			entered in an invalid
				form!

3	Некорректны й ввод	E+54	-	ERROR: a real number is entered in an invalid
				form!
4	Количество цифр в мантиссе > 30	99999999999999999999999999999999999999	-	ERROR: the number of digits in the mantissa must not exceed 30!
5	Количество цифр в мантиссе > 30 при наличии точки	+9999999999999999999999999999999999999	-	ERROR: the number of digits in the mantissa must not exceed 30!
6	Количество цифр в порядке > 5	+13E+555555	-	ERROR: the number of digits in the order must not exceed 5!
7	Порядок – не целое число	+1234E+54.7	-	ERROR: a real number is entered in an invalid form!
8	Некорректны й ввод	+15.45E+54	15.53.23	ERROR: a real number is entered in an invalid form!
9	Деление на ноль	+15.45E+54	+0.0	ERROR: division by zero!
1	Деление нуля на число	+0.0	+15.45E+54	Result: +0.0E+0
1	Деление нуля на ноль	+0.0	+0.0	ERROR: division by zero!
1 2	Деление максимально возможного числа на единицу	+9999999999999999999999999999999999999	+1	ERROR: power was overflow!
1 3	Деление максимально возможного числа на минимально возможное	+9999999999999999999999999999999999999	- 9999999999999999999999 99999999.E-99999	ERROR: power was overflow!
1 4	Деление минимально возможного числа на максимально возможное	- 999999999999999999999999999999999999	+9999999999999999999999999999999999999	ERROR: power was overflow!
1 5	Округление	+2	+3	Result: +0.666666666666666666666666666666666666
1 6	Деление чисел с разными знаками	+15	-2	Result: - 0.75000000000000000000 000000000000E+1
1 7	Максимально возможный порядок	+1000.45	+1000.45E+99999	Result +0.10000000000000000000 000000000000000
1 8	Проверка на правильность нормализаци и полученного числа	+0.01E-99999	+2	ERROR: power was overflow!

1	Проверка на	+999999999999999999	+2	Result
9	округление с	99999999		+0.50000000000000000000
	поразрядным			00000000000E+30
	переносом с			
	коррекцией			
	порядка			

6. Выводы

Для представления больших чисел в памяти компьютера очень удобно использовать такую структуру данных, как структуру. Это позволяет хранить число по частям и удобно оперировать ими.

7. Ответы на контрольные вопросы

7.1. Каков возможный диапазон чисел, представляемых в ПК?

Диапазон чисел напрямую зависит от области памяти, выделяемый под хранение переменной, наличия знака, а также типа представления числа (вещественное или целое). К примеру, если целая беззнаковая переменная типа int занимает 4 байта (16 бит), то диапазон её значений: 0..2^16-1 - 0..65535. Если же она может принимать также и положительные значения, то диапазон: -2^15..2^15-1

7.2. Какова возможная точность представления чисел, чем она определяется?

Вещественные числа хранятся в представлении с плавающей точкой в виде $X = M * E^p$, где M — мантисса, представленная правильной дробью (в интервале [0.1..1)). Точность представления вещественного числа зависит от максимально возможной длины мантиссы, которая, зависит от области выделяемой памяти и наличия знака. Если длина мантиссы выходит за границы разрядной сетки, то происходит округление.

Таким образом, если под хранение целого положительного числа

выделено 16 разрядов, то его максимальное значение не может превышать 2^{16} -1=65 535, если выделено 32 разряда, то максимальное значение составит 2^{32} -1=4 294 967 295. Для 64 разрядов максимально возможное значение числа равно 2^{64} - 1=18 446 744 073 709 551 615.

7.3. Какие стандартные операции возможны над числами?

Стандартные операции над числами – сложение, вычитание, умножение, деление.

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

Для представления больших чисел в памяти компьютера можно использовать массив символов char, но это не совсем удобный способ представления, если необходимо с такими числами совершать арифметические операции. Более удобный способ – представление числа в виде структуры, хранящей в себе мантиссу, порядок (если он есть), знак числа, и т.д.

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

Операции над большими числами можно выполнять путём последовательного выполнения операций над всеми цифрами, составляющими число, начиная с «хвоста» массива цифр.