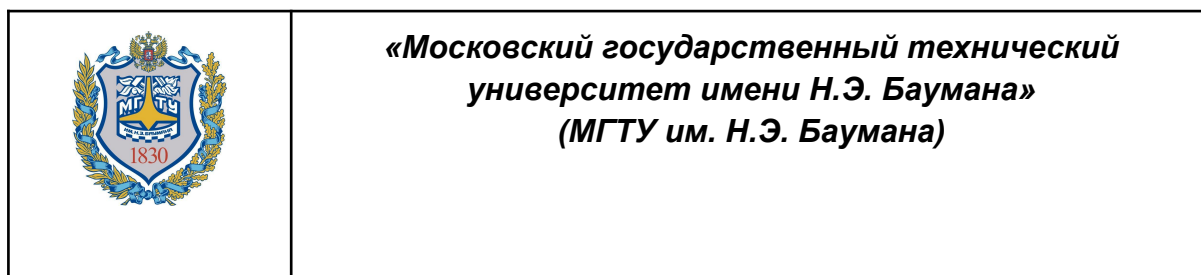


Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования



ФАКУЛЬТЕТ _____ ИУ _____

КАФЕДРА _____ ИУ7 _____

Отчет

по лабораторной работе № 1

Дисциплина: Типы и Структуры Данных

Название лабораторной работы: Обработка больших чисел

Студент гр. **ИУ7-34Б Малышев Илья Николаевич**

(Подпись, дата) (И.О. Фамилия)

Преподаватель **Силантьева Александра Васильевна**

(Подпись, дата) (И.О. Фамилия)

Москва, 2021

Условия задачи

Необходимо составить программу деления двух чисел, где порядок имеет до 5 знаков: от -99999 до $+99999$, а мантисса – до 30 знаков. Программа должна осуществлять ввод чисел и выдавать либо верный результат в указанном формате (при корректных данных), либо сообщение о невозможности произвести счет.

Техническое задание

Формат входных данных: На вход от пользователя программа принимает две строки длиной не более 64 символов. В первой строке записано целое число, длина мантиссы которого не превышает 30. Во второй строке записано действительное число в произвольном формате, то есть научном или естественном, длина мантиссы также не превышает 30 знаков, модуль порядка числа не превышает 99999. Все символы, кроме арабских цифр, знака минус, точки и букв 'e' и 'E', запрещены.

Формат выходных данных: строка, содержащая информацию об успешности (неуспешность) выполнения программы, строка содержащая результат записанный в экспоненциальном формате, длина мантисс не превышает 30 знаков, модуль порядка не превышает 99999.

Задача программы: произвести деление чисел, значения которых не помещаются в стандартные типы языка Си, при невозможности - вернуть сообщение об ошибке.

Способ обращения к программе: программа является консольной. После запуска исполняемого файла из консоли, программа выведет справку и приглашение на ввод.

Аварийные ситуации (ошибки пользователя):

1. Пользователь ввёл недопустимые символы;
2. Пользователь ввёл число, длина мантиссы которого превышает 30;

3. Пользователь не ввёл число;
4. Пользователь ввёл действительное число, когда нужно было вводить целое;
5. Пользователь ввёл число модуль порядка которого превышает 99999;
6. В результате выполнения программы было получено число, модуль порядка которого превышает 99999.

Внутренние структуры данных

Помимо стандартных типов данных, в программе используются два пользовательских: `big_int_t` и `big_real_t`.

`big_int_t` : представляет собой модель целого числа с 30-ти символьной мантиссой. В программе задан структурой из двух полей:

`int is_negative;`

`unsigned char mantissa[_MANTISSA_LEN_].`

Поле `is_negative` является булевым и хранит информацию о знаке числа. В массиве `mantissa` посимвольно хранится мантисса числа.

`big_real_t`: представляет собой модель действительного числа с 30-ти символьной мантиссой. В программе задан структурой из двух полей:

`big_int_t big_int;`

`int exponent.`

В поле `exponent` хранится порядок числа. В поле `big_int` хранится целая часть числа.

Описание алгоритма

- Считывание “длинного” целого, если возникает ошибка: вывод сообщения о ней, выход;
- Считывание “длинного” вещественного, если возникает ошибка: вывод сообщения о ней, выход;
- Приведение “длинного” целого к “длинному” вещественному;
- Деление целых составляющих чисел;

- Получение результата в виде “длинного” вещественного числа;
- Проверка результата на корректность, если он некорректен:
вывод сообщения об ошибке, выход;
- Вывод результата;
- Выход.

Тесты

Негативные:

- Проверка корректности ввода

Большое целое	Большое действительное
q	1
1	q
q	q
1.1	1.q
.1	1
1.	1
11111111111111111111111111111111	1
1	11111111111111111111111111111111
1	1e111111
1	1eq
1	1,1
	1
1	

- Проверка результата

100000	1y-99999
--------	----------

Позитивные:

Большое целое	Большое действительное	Результат
1	2	0.5e0
2	1	0.2e1
999999999999999999 999999999999999999	2	0.5e30
-1	2	-0.5e0
1	0.125	0.8e1
1	0.8e1	0.125e0
101	10.1e1	0.1e1

Выводы по проделанной работе

При обработке “больших” чисел, диапазон значений которых выходит за диапазоны значений стандартных типов данных, разработчик вынужден проектировать собственный тип данных, который отвечал бы поставленным условиям задачи. При проектировании необходимо написать некоторое количество функций, которые позволили бы программисту пользоваться введенным типом данных так, будто это стандартный тип данных, с некоторыми оговорками.

Программа была написана. Требованиям отвечает.