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 «Программное обеспечение ЭВМ и информационные технологии»

НАПРАВЛЕНИЕ ПОДГОТОВКИ «09.03.04 Программная инженерия»

# ОТЧЕТ

по лабораторной работе №1

Название: Длинная арифметика

Дисциплина: Типы и структуры данных

Вариант: 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент: | ИУ7-32Б |  | 28 сентября 2021 г. | А. П. Лемешев |
|  | группа | подпись | дата | (И. О. Фамилия) |
| Преподаватель: |  | подпись | дата | А.В. Силантьева  (И. О. Фамилия) |

Москва — 2021 г.

**Условие задачи**  
Составить программу умножения действительного числа на целое, где порядок у действительного имеет до 5 знаков: от –99999 до +99999, а мантисса у обоих чисел – до 30 знаков.

**Техническое задание**Смоделировать операцию умножения действительного числа в форме   
+-m.n Е +-K, где суммарная длина мантиссы (m+n) - до 30 значащих цифр, а величина порядка K - до 5 цифр, на целое число длиной до 30 десятичных цифр. Результат выдать в форме +-0.m1 Е +-K1, где m1 - до 30 значащих цифр, а K1 - до 5 цифр.

**Входные данные:**1. Строка, в которой записано действительное число в экспоненциальной или обычной форме. Перед мантиссой и порядком обязательно стоит знак (+ или -).  
2. Строка в которой записанно целое число. Запись в экспоненциальной форме запрещена. Перед мантиссой обязательно стоит знак (+ или -).

**Формат входных данных:**Примеры ввода действительного числа:  
a) +65.23E+10 // обычный ввод  
b) -65.23E-10 // знак минус  
c) +23E+5 // как целое число  
d) +23.72 // без порядка  
e) +.102E+1 // до точки нет числа  
f) +85.E+8 // после точки нет числа  
Примеры ввода целого числа:  
a) +75 // обычный ввод  
b) -65 // знак минус  
c) +0002 // незначащие нули

**Выходные данные:**  
Строка, в которой записан результат посимвольного перемножения этих двух cтрок в нормализованном формате.

**Формат выходных данных:**  
{знак}0.{мантисса не больше 30 символов}e{знак}{порядок}  
Пример: +0.3871142099076e+21

**Функция программы:**  
Умножение действительного числа на целое.

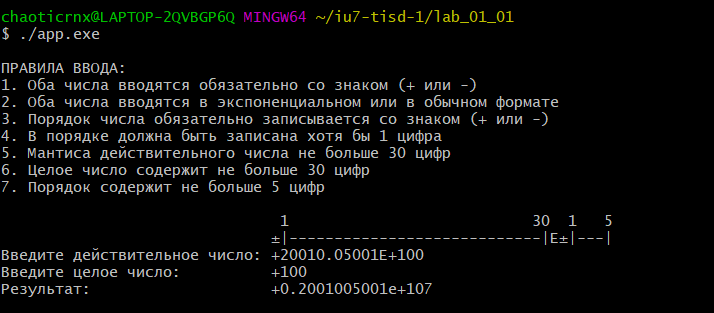
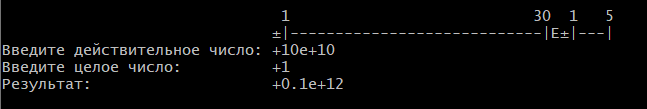
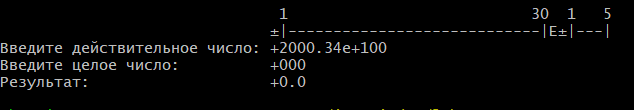
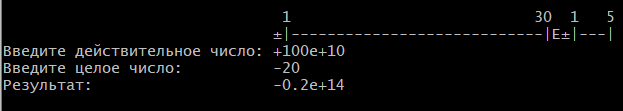
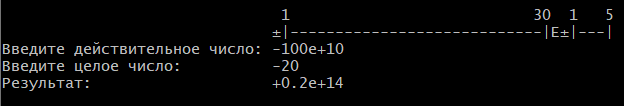
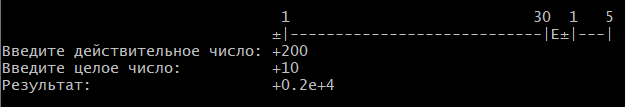
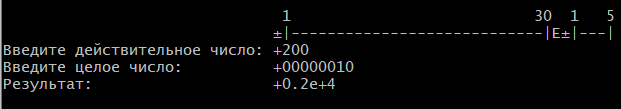
**Обращение к программе:**  
Запускается в консоли.

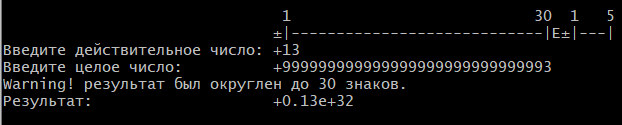
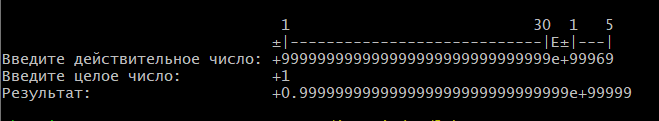
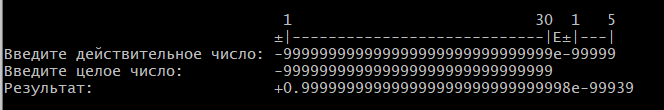
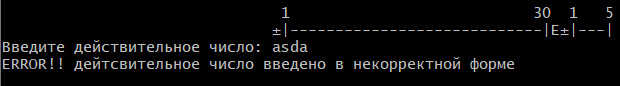
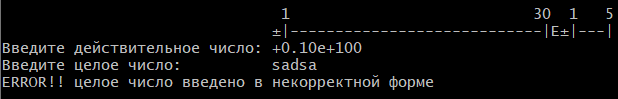
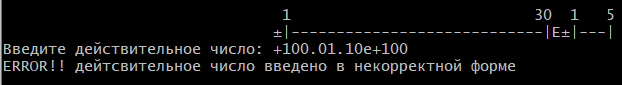
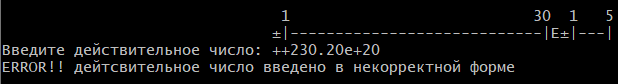
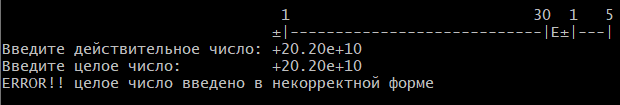
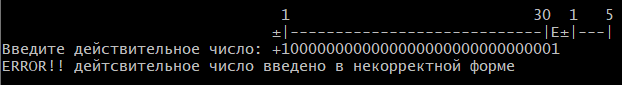
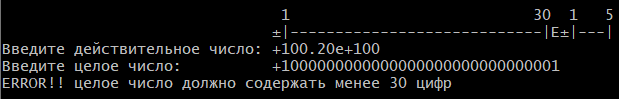
**Возможные ошибки:**  
1. Ошибка ввода: неправильная запись или превышение длины числа.  
2. Превышение допустимого порядка при округлении числа.

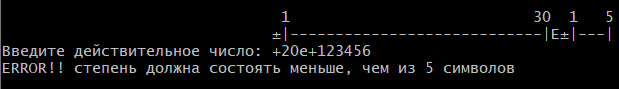
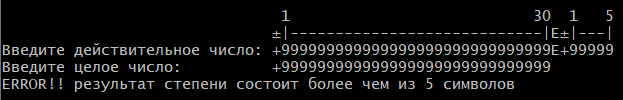
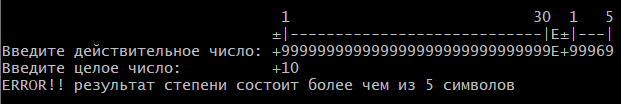
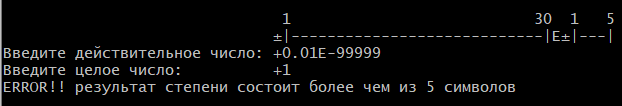
**Структура данных**  
В программе используется только одна структура данных, это запись (структура) с именем bignum\_t.  
typedef struct  
{  
 char mantissa\_sign;  
 char mantissa[MAX\_MANTISSA \* 2];  
 int degree;  
 litnum\_t point\_ind;  
} bignum\_t;  
В данной СД есть четыре поля: первое отвечает за знак мантиссы, во втором посимвольно хранится мантисса, в третьем поле хранится порядок целым числов, последнее отвечает за хранение индекса точки. В данную СД записывается первое, второе входное число и сам результат вычислений.

**Алгоритм**

1. Ввод действительного числа  
   a) приведение к эспоненциальному виду;  
   b) обработка ошибок при вводе;  
   c) конвертация введеной строки в структура;
2. Ввод целого числа  
   a) приведение к эспоненциальному виду;  
   b) обработка ошибок при вводе;  
   c) конвертация введеной строки в структура;
3. Перемножение чисел  
   a) берем последнюю цифру целого числа;  
   b) умножаем на эту цифру мантиссу действительного числа;  
   c) результат добавляем в буфер;  
   d) после прохода по всему целому числу (от конца до начала) и прибавления результатов в буфере будет результат умножения;
4. Округление результата и обработка ошибок
5. Вывод результата

**Тесты  
Позитивные тесты**Обычный тест  
  
Умножение на 1  
  
Умножение на 0  
  
Умножение на минус  
  
  
Умножение целых чисел  
  


Округление  
  
Пограничные значения  
  
  
**Негативные тесты**Неправильный ввод  
  
  
  
  
  
Превышена длина мантиссы  
  


Превышена длина порядка  
  
Переполнение  
  
  


**Контрольные вопросы:**

1. **Каков возможный диапазон чисел, представляемых в ПК?**  
   Диапазон значений чисел зависит от разрядности процессора, размера выделенной для хранения переменной памяти, от типа (целое/вещественное), знака. Под хранение целого положительного числа, для которого выделено 64 разряда, максимально возможное значение числа равно 2^64 – 1 = 18 446 744 073 709 551 615.
2. **Какова возможная точность представления чисел, чем она определяется?**Точность представленяи вещественного числа зависит от максимально возможной длины мантиссы, которая, опять-таки, зависит от области выделяемой памяти и наличия знака. Если длина мантиссы выходит за границы разрядной сетки, то происходит округление. Обычно, длина мантисы это 20 десятичных разрядов.
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 использовать специальные библиотеки для работы с большими числами; создать массив цифр числа и производить операции поэлементно, храня промежуточный результат в отдельном массиве; создать структуру, деля в ней мантису на части.
5. **Как можно осуществить операции над числами, выходящими за рамки машинного представления?**Сохранить число в массив (символов или интов), и производить действия по элементно (над каждой цифрой), например, используя классический алгоритм “столбиком”

**Вывод по проделанной работе**При реализации аримфетики длинных чисел целесообразно хранить цифры числа, как элементы массива. Арифметику производить поразрядно.