



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

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

«Длинная арифметика»

Студент _____ Маслова Марина Дмитриевна
фамилия, имя, отчество

Группа _____ ИУ7-23Б

2020 г.

Оглавление

Техническое задание	3
Условие задачи	3
Входные данные	3
Выходные данные	3
Задача, реализуемая программой	3
Способ обращения к программе	3
Возможные аварийные ситуации и ошибки пользователя	3
Описание внутренних структур данных	4
Описание функций	5
Описание алгоритма	6
Тесты.....	7
Контрольные вопросы	9
Вывод.....	10

Техническое задание

Условие задачи

Смоделировать операцию умножения действительного числа в форме $\pm m.n \text{ E } \pm K$, 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 $\pm 0.m1 \text{ E } \pm K1$, где $m1$ – до 30 значащих цифр, а $K1$ – до 5 цифр.

Входные данные

1. Строка, в которой записано длинное вещественное число в экспоненциальной или обычной форме. Строка должна иметь вид $\pm m.n \text{ E } \pm K$, где суммарная длина мантиссы ($m+n$) – до 30 значащих цифр, а величина порядка K – до 5 цифр при этом ввод точки, экспоненты, пробелов и знака «+» необязателен.

2. Строка, в которой записано длинное целое число. Строка должна иметь вид $\pm n1$, где длина $n1$ не превышает 30 при этом ввод знака «+» необязателен.

Выходные данные

Строка, являющаяся результатом операции умножения двух введенных чисел и представленная в виде $\pm 0.m1 \text{ E } \pm K1$, где $m1$ – до 30 значащих цифр, а $K1$ – до 5 цифр.

Задача, реализуемая программой

Осуществление операции умножение длинного вещественного числа на длинное целое число.

Способ обращения к программе

Через терминал.

Возможные аварийные ситуации и ошибки пользователя

Некорректный ввод (использование недопустимого символа, превышение максимально возможного количества символов, неверный формат ввода).

Переполнение порядка в результате умножения.

Описание внутренних структур данных

В программе используются две структуры данных: одна – для представления длинного вещественного числа – `big_double`, другая – для представления длинного целого числа – `big_int`.

typedef struct

```
{  
    char sign;  
    char num[MAX_DOUBLE_LEN];  
    short int len_num;  
    short int point_place;  
    int order;  
} big_double;
```

Структура `big_double` имеет 5 полей:

char sign – знак вещественного числа;

char num[MAX_DOUBLE_LEN] – массив символов для хранения мантиссы числа;

short int len_num – длина мантиссы;

short int point_place – положение точки отсчитывая от старшего разряда;

int order – порядок вещественного числа.

В данную структуру записывается введенное вещественное число и результат вычислений.

typedef struct

```
{  
    char num[MAX_INT_LEN + 1];  
    short int len_num;  
} big_int;
```

Структура `big_int` имеет два поля:

char num[MAX_INT_LEN + 1] – массив символов, представляющих цифры числа и знак, хранящийся в элементе с индексом 0.

short int len_num – длина целого числа.

В данную структуру записывается введенное целое число.

Описание функций

`mulriply_big_numbers()`

Осуществляет умножение длинного вещественного и длинного целого чисел

Parameters

- [in] **int_num** Указатель на длинное целое число
- [in] **double_num** Указатель на длинное вещественное число
- [out] **result_num** Указатель на результат (длинное вещественное число)

Returns

Код ошибки

`normalize_number()`

Преобразовывает длинное вещественное число к нормализованному виду

Parameters

- [in, out] **Указатель** на длинное вещественное число

`read_big_double()`

Считывает длинное вещественное число.

Parameters

- [out] **number** Указатель на длинное вещественное число

Returns

Код ошибки

`print_big_double()`

Выводит нормализованное вещественное число на экран

Parameters

- [in] **number** Указатель на длинное вещественное число

`read_int_str()`

Считывает строку для последующего преобразования в int.

Parameters

- [in] **max_int_len** Максимальное количество цифр в числе
- [out] **read_str** Указатель на считанную строку

Returns

Код ошибки

`read_mantissa()`

Считывает мантиссу длинного вещественного числа.

Parameters

- [out] **number** Указатель на длинное вещественное число

Returns

Код ошибки (целое число)

`read_order()`

Считывает порядок длинного вещественного числа.

Parameters

- [out] **order** Указатель на порядок вещественного числа

Returns

Код ошибки

`round_num()`

Округляет вещественное число при выходе за разряды мантиссы

Parameters

- [in] **last** Последняя вышедшая за разряды цифра
- [out] **number** Указатель на округленное вещественное число

read_big_int()

Считывает длинное целое число

Parameters

[out] **number** Указатель на длинное целое число

Returns

Код ошибки

Описание алгоритма

Для осуществления операции умножения используется алгоритм вычисления методом Транхенберга. Он представляет собой модифицированный способ умножения «столбиком», в котором каждая цифра числа вычисляется сразу, что не требует отдельного хранения результатов умножения каждой цифры второго числа на первое.

Тесты

№	Что проверяется	Вещественное число	Целое число	Результат
1	Некорректный формат ввода	ert	--	Ошибка при чтении порядка
2	Некорректный формат ввода	iu7	--	При вводе использован некорректный символ
3	Некорректный формат ввода	1e23	FGH	При вводе использован некорректный символ
4	Некорректный формат ввода	-12.3e+12.3	--	Ошибка при чтении порядка
5	Некорректный формат ввода	--1e+21	--	При вводе использован некорректный символ
6	Некорректный формат ввода	1e88	-1e56	При вводе использован некорректный символ

22	Положительное на отрицательное	+12	-12	-0.144 E +3
23	Отрицательное на положительное	-12	+12	-0.144 E +3
24	Отрицательное на отрицательное	-12	-12	+0.144 E +3

Контрольные вопросы

Каков возможный диапазон чисел, представляемых в ПК?

Если n – число разрядов процессора, то:

для целых положительных чисел: $0 < x < 2^n - 1$

для целых отрицательных чисел: $-2^{n-1} \leq x < 0$

для вещественных чисел: $3.6E-4951 \leq x \leq 1.1E+4932$ (максимальный размер мантиисы 52 двоичных разряда, порядка – 11 разрядов).

Какова возможная точность представления чисел, чем она определяется?

Точность представления вещественных чисел зависит от максимального количества разрядов, отведенных под хранение мантиисы. При выходе мантиисы из разрядной сетки происходит округление и точность теряется. Под хранение мантиисы числа типа double отводится 52 двоичных разряда, что соответствует не более, чем 20 десятичным разрядам.

Какие стандартные операции возможны на числами?

Сложение, вычитание, умножение, деление, сравнение, деление нацело, взятие остатка.

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

Самым выгодным по памяти вариантом является массив элементов типа char. Так же можно использовать целые типы (int, short int), однако по

сравнению с массивом символов массив целых чисел будет занимать больше памяти.

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

Производить действия над числами поэлементно, заранее сохранив цифры числа в массив. Для выполнения операций использовать алгоритм «столбиком или его модификации.

Вывод

В ходе выполнения лабораторной работы был успешно реализован алгоритм умножения длинного вещественного числа на длинное целое число, были получены навыки хранения чисел, не входящих в диапазон представления в ПК, и работы с ними.

Было выяснено, что для хранения чисел, выходящих за рамки машинного представления, эффективно использовать структуры для вещественных чисел – с полями, соответствующими знаку (символьный тип), мантиссе (массив символов), её длине (короткое целое), порядку (целое) и положению точки (короткое целое), для целых – самому числу со знаком (массив символов) и его длине (короткое целое).