



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 Тузов Даниил Александрович

Группа ИУ7 – 32Б

Преподаватель Силантьева Александра Васильевна
Барышникова Марина Юрьевна

Оглавление

| | |
|---|----------|
| <u>ОПИСАНИЕ УСЛОВИЯ ЗАДАЧИ.....</u> | <u>3</u> |
| <u>ОПИСАНИЕ ТЕХНИЧЕСКОГО ЗАДАНИЯ.....</u> | <u>3</u> |
| <u>ОПИСАНИЕ СТРУКТУРЫ ДАННЫХ.....</u> | <u>4</u> |
| <u>ОПИСАНИЕ АЛГОРИТМА.....</u> | <u>5</u> |
| <u>ОПИСАНИЕ ФУНКЦИЙ.....</u> | <u>5</u> |
| <u>НАБОР ТЕСТОВ.....</u> | <u>6</u> |
| <u>ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ.....</u> | <u>8</u> |
| <u>ВЫВОД.....</u> | <u>8</u> |

ОПИСАНИЕ УСЛОВИЯ ЗАДАЧИ

Смоделировать операцию деления действительного числа в форме $\pm m.n E \pm K$, где суммарная длина мантиссы ($m+n$) - до 35 значащих цифр, а величина порядка K - до 5 цифр, на целое число длиной до 35 десятичных цифр. Результат выдать в форме $\pm 0.m_1 E \pm K_1$, где m_1 - до 35 значащих цифр, а K_1 - до 5 цифр.

ОПИСАНИЕ ТЕХНИЧЕСКОГО ЗАДАНИЯ

Входные данные:

Действительное число в формате $\pm m.nE \pm K$, где $\text{len}(n) + \text{len}(m) \leq 35$ и $99999 \leq K \leq 99999$

И целое число в формате $\pm K$, где $\text{len}(K) \leq 35$

Выходные данные:

Действительное число в формате $\pm m.nE \pm K$, где $\text{len}(n) + \text{len}(m) \leq 35$ и $99999 \leq K \leq 99999$

Описание программы:

Деление действительного числа на целое

Способ обращения к программе:

Запускается через терминал. Сначала вводится действительное число, а потом целое

Аварийные ситуации:

1. Некорректный ввод: переполнения буфера при считывании строки, в которой записано наше число
Код ошибки – 5.
2. Некорректный ввод: не удалось считать строку в буфер
Код ошибки – 1.
3. Некорректный ввод: строка не соответствует формату действительного числа
Код ошибки – 2.
4. Некорректный ввод: строка не соответствует формату целого числа
Код ошибки – 3.
5. Некорректный ввод: слишком много цифр в мантиссе действительного числа или же в порядке действительного числа

или же слишком много цифр в записи целого числа

Код ошибки – 4.

6. Невозможность деления (деление на ноль)

Код ошибки – 6.

7. Переполнение порядка в ответе

Код ошибки – 7.

ОПИСАНИЕ СТРУКТУР ДАННЫХ

Для считывания чисел из консоли используются строки. Затем числа, хранимые в строках, записываются в соответствующие структуры

Структура действительного числа:

```
struct real_t
{
    bool mant_sign;
    int mant[MANT_LEN];
    size_t mant_len;
    bool exp_sign;
    int exp;
};
```

Поля:

mant_sign — знак мантиссы

mant — мантисса

mant_len — длина мантиссы

exp_sign — знак порядка

exp — значение порядка

Структура целого числа:

```
struct int_t
{
    bool sign;
    int mant[MANT_LEN];
    size_t mant_len;
};
```

Поля:

sign – знак числа

mant — само число

mant_len — количество цифр числа

ОПИСАНИЕ АЛГОРИТМА

1. Программа считывает две строки, одна содержит действительное число, другая — целое
2. Строки проверяются на корректность записи действительного числа и целого числа
3. Проверяется корректность количества цифр в записе чисел
4. Числа записываются в соответствующие структуры (из записи чисел убираются ведущие нули, действительное число сразу нормализуется)
5. Выполняется деление чисел “в столбик”
6. Результат записывается в новую переменную в нормализованном виде
7. Результат выводится на экран

ОПИСАНИЕ ФУНКЦИЙ

`int read_string(FILE *f, char *str);` - функция, которая читает из потока `f` строку `str`

`bool is_correct_exp_val(const char *str, size_t *col_digit_m, size_t *col_digit_k);` - функция, которая проверяет на корректность вещественное число записанное в строке `str` и записывает количество цифр в мантиссе и в показателе в соответствующие переменные `col_digit_m` и `col_digit_k`

`bool is_correct_int_val(const char *str, size_t *col_digit);` - функция, которая проверяет на корректность целое число записанное в строке `str` и записывает количество цифр в `col_digit`

`void convert_real(const char *real_string, real_t *val);` - функция, которая конвертирует строку `real_string`, в которой записано вещественное число, в структурную переменную вещественного числа (записывает в нормализованном виде)

`void convert_int(const char *int_string, int_t *val);` - функция, которая конвертирует целое число, записанное в `int_string`, в специальную структурную переменную этого целого числа

`int compare(const int arr1[], size_t n1, const int arr2[], size_t n2);` - функция сравнения двух длинных чисел, цифры которых записаны в массивах `arr1` и `arr2` соответственно

`void subtraction(int arr1[], size_t *n1, const int arr2[], size_t n2);` - функция вычитания из двух длинных чисел (результат сохраняется в arr1)

`int division(real_t *dividend, int_t *divider, real_t *res);` - функция, деления вещественного числа dividend на целое число divider, результат записывается в res

`void print_res(real_t *res);` - функция печати вещественного числа

НАБОР ТЕСТОВ

| № | Описание теста | Действительное | Целое | Результат |
|----|--|--|---|-----------|
| 1 | Буква вместо действительного числа | a | 1 | Ошибка |
| 2 | Буква вместо целого числа | 1e1 | a | Ошибка |
| 3 | Слишком большая мантисса | 11111111111111 11111111111111 11111111111111 11111111111111 11111111111111 11111E23 | -1 | Ошибка |
| 4 | Слишком большой порядок | 1e23232323 | 23 | Ошибка |
| 5 | Слишком много цифр в записи целого числа | 1e1 | 123333333333 33333333222 22222222222 21111111111 11112222222 223333333 | Ошибка |
| 6 | Деление на ноль | 1 | 0 | Ошибка |
| 7 | Два знака в записи действительного числа | --12 | 3 | Ошибка |
| 8 | Две точки в записи действительного | 12..E4 | 5 | Ошибка |
| 9 | Экспонента без значения | 12E | 1 | Ошибка |
| 10 | Ни одной значащей цифры | .E2 | 1 | Ошибка |

| | | | | |
|----|---|--|--|------------|
| 11 | Два знака в записи целого | 2 | --1 | Ошибка |
| 12 | Точка в записи целого | 12 | 12.3 | Ошибка |
| 13 | Экспонента в записи целого | 12 | 12E1 | Ошибка |
| 14 | Переполнение порядка в ответе | 99999e99999 | 1 | Ошибка |
| 15 | Переполнение порядка в нижнюю сторону | 0.001e-99999 | 1 | Ошибка |
| 16 | Деление целого на целое | 1 | 1 | 0.1E1 |
| 17 | Деление вещественного на целое | 1E-35 | 1 | 0.1E-34 |
| 18 | Деление отрицательного на отрицательное | -111 | -1 | 0.111E3 |
| 19 | Деление отрицательного на положительное | -121 | 11 | -0.11E2 |
| 20 | Деление отрицательного действительного на положительное | -121.121 | 11 | -0.11011E2 |
| 21 | Граничный тест: деление на очень большое | 0.1 | 100000000000 000000000000 000000000000 | 0.1E-34 |
| 22 | Ноль делится на что-то | 0 | 10000 | 0.E1 |
| 23 | Числа с ведущими нулями | 0000000000000000 0000000000000000 0000000000011.11 | 000000000000 000000000000 000000000000 0011 | 0.101E1 |
| 24 | Тест на цепочное округление | 99999999999999 99999999999999 999999 (35 девяток) | 2 | 0.5E35 |

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Каков возможный диапазон чисел, представляемых в ПК?

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 $2^{64}-1=18\,446\,744\,073\,709\,551\,615$

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

Длина мантиссы определяет точность представления числа, а длина порядка ограничивает диапазон допустимых значений. Максимально под представление мантиссы отводится 52 двоичных разряда, а под представление порядка – 11 двоичных разрядов. В этом случае возможные значения чисел

находятся в диапазоне от $3.6\text{ E }-4951$ до $1.1\text{ E }+4932$

3. Какие стандартные операции возможны над числами?

Сложение, вычитание, умножение, деление, взятие остатка, сравнение, битовые сдвиги

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

Массив символов/структуру, одно из полей которой массив символов

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

С помощью длинной арифметики. Когда реализация стандартных операций ложится на плечи разработчика, который в свою очередь может реализовать длинную арифметику на специальных типах данных

Вывод

В ходе этой работы я изучил понятие длинной арифметики — методологию представления чисел, выходящих за рамки машинного представления, в виде массивов символов и структур. Этот способ оперирования с длинными числами довольно непростой в использовании, потому что написанные функции оказываются довольно тяжело

читаемыми в силу того, что приходится учитывать множество разных случаев