Лабораторная работа № 1

«Длинная» арифметика.

Выполнил Елгин И.Ю.

Группа ИУ7-34Б

**Задача**:

Смоделировать операцию деления целого числа длиной до 30 десятичных цифр на действительное число в форме ±m.n Е ±K, где суммарная длина мантиссы (m+n) - до 30 значащих цифр, а величина порядка K - до 5 цифр. Результат выдать в форме ±0.m1 Е ±K1, где m1 - до 30 значащих цифр, а K1 - до 5 цифр.

**Формат входных данных:**

Пользователь вводит в программу через консоль два числа. Числа вводятся по одному в строку. Для ввода числа используется символ новой строки. Первое число – целое в диапазоне от -99999999999999999999999999999 до 9999999999999999999999999999 (ввод любых символов, кроме цифр и знаков “+”,”-” в начале , считается ошибкой), второе число – вещественное, вводится в формате m.n Е K (m+n) - до 30 значащих цифр K - до 5 цифр , при вводе вещественного числа требуется ввод “E” или “e” и указанием степени 10. Пустой ввод рассматривается как ошибка ввода.

**Формат выходных данных:**

Результат деления выводится в формате 0.m1 Е K1, где m1 - до 30 значащих цифр, а K1 - до 5 цифр. Результат деления выводится с точностью до 30-ти знаков ,30 знак округляется. Если в процессе деления произошла ошибка, то сообщение об этом выводится на экран, и программа завершается с кодом ошибки.

**Аварийное завершение программ**:

-Если второе введённое число равно 0, то программа выдаст сообщение “Попытка деления на 0!” и экстренно завершится с кодом ошибки 1.

-Если ввод чисел не будет соответствовать ТЗ, то программа выдаст сообщение “Ввод не соответствует формату” и завершится.

-Если произошло переполнение порядка в результате деления, то программа выдаст сообщение “Переполнение” и завершится.

**Обращение к программе**:

Исполняемый файл main.exe . Запускается из командной строки, аргументы строки не требуются.

**Структуры данных:**

char ch2[96];

Для целого числа используется массив символов в первом элементе содержится знак, в других значащие цифры от 0 до 9 или 10, 10 – значит, что элемент и все последующие фактически = 0.

struct my\_float

{

char digit[34];

long long e;

};

Для делителя используется массив символов в первом элементе содержится знак, второй нулевой и используется для сдвига, в других значащие цифры от 0 до 9 или 10, 10 – значит, что элемент и все последующие фактически = 0 для степени 10 используется переменная типа long long (64 бита).

struct my\_float

{

char digit[34];

long long e;

};

Для результата используется массив символов в первом элементе содержится знак, в других значащие цифры от 0 до 9 или 10, 10 – значит, что элемент и все последующие фактически = 0 для степени 10 используется переменная типа long long (64 бита).

**Алгоритм**:

Если формат вводимых чисел корректный, то

Записать числа в массив и нормализовать их

Если делитель не является нулём то

До тех пор, пока нет 31 сдвигов делать

Если можно вычесть делитель из делимого то

Делимое минус делитель

Цифра результата с номером сдвига + 1

Иначе сдвиг.

Отформатировать результат

Если нет переполнения, то

Вывести результат.

Иначе вывести сообщение о переполнении.

Иначе вывести сообщение об ошибке

Иначе вывести сообщение о некорректном вводе.

**Функции**:

*Int input\_int*(char \*c, int \*zn)

- Функция реализует ввод целого числа (до 30-ти знаков) с консоли.

- Параметры на вход: c-массив в который записывается число, zn – количество знаков в числе.

- Возвращает: код ошибки или 0 в качестве результата, вещественное число, записанное в массив.

*Int input\_float*(char \*c, long long \*e)

- Функция реализует ввод вещественного числа (до 30-ти знаков) с консоли.

- Параметры на вход: c-массив в который записывается число, e –степень множителя 10.

- Возвращает: код ошибки или 0 в качестве результата, вещественное число, записанное в массив c и его порядок e.

*void print\_result*(char \*res, long long e)

- Функция реализует вывод результата деления.

- Параметры на вход: res-массив в который записывается результат деления, e –степень множителя 10.

- Ничего не возвращает.

*int division*(char \*\*delim, char \*\*delit, char \*result, long long \*e)

- Функция реализует деление.

- Параметры на вход: delim-массив в который записывается делимое число, delit -массив в который записывается делитель, result-массив для результата, e –степень множителя 10.

- Возвращает: код ошибки или 0 в качестве результата, вещественное число – результат деления, записанное в массив result и его порядок e.

*int shift*(char \*\*ch1, char \*\*ch2)

- Функция реализует сдвиг делимого или делителя относительно друг друга.

- Параметры на вход: c1-массив в который записывается делимое число, c2-массив в который записывается делитель.

- Возвращает: 1 в случае если сдвиг произведён и 0 в ином случае в качестве результата, и числа, записанные в массивы ch1, ch2.

*void minus*(char \*ch1, char \*ch2)

- Функция реализует вычитание одного числа из другого.

- Параметры на вход: ch1-массив в который записывается число и которого вычитают, ch2 – массив в который записывается вычитаемое число.

-Изменяет число ch1 из которого вычли число ch2.

*int compere*(char \*ch1, char \*ch2)

- Функция реализует сравнивает два числа.

- Параметры на вход: ch1-массив в который записывается 1 число, ch2-массив в который записывается 2 число.

- Возвращает 1 если первое число больше второго и 0 в ином случае.

*void correct\_result*(char \*res, char \*c1, char \*c2, long long \*e)

- Функция приводит результат к правильному формату, устанавливает знак, округляет, убирает незначащие 0.

- Параметры на вход: res-массив в который записывается результат, с1-массив в который записывается делимое число, с2 -массив в который записывается делитель, e –степень множителя 10.

-Изменяет результат в форме для вывода, записанный в массив res и его порядок e.

**Тесты**:

|  |  |  |  |
| --- | --- | --- | --- |
| № | Ввод | Случай | Вывод |
| 1 | 1  0e0 | Деление на 0 | Попытка деления на 0! (Выход с кодом 1) |
| 2 | 0  99.99e10 | Деление нуля | 0 |
| 3 | 0  0e0 | Деление 0 на 0 | Попытка деления на 0! (Выход с кодом 1) |
| 4 | 2  56.1 | Некорректный ввод | Ввод не соответствует формату (Выход с кодом 2) |
| 5 |  | Пустой ввод | Ввод не соответствует формату (Выход с кодом 2) |
| 6 | 999999999999999999999999999999999999999999 | Ввод более 30ти знаков | Ввод не соответствует формату (Выход с кодом 2) |
| 7 | 214  2e-1 | Стандартное целое деление | +0.107E4 |
| 8 | 999999999999999999999999999999  1e0 | Стандартное целое деление | +0.999999999999999999999999999999E30 |
| 9 | 1  3e0 | Стандартное дробное деление без округления | +0.333333333333333333333333333333E0 |
| 10 | 54389534526563  34656.3452345e45345 | Стандартное деление | 0.156939614256897559646221269524E-45335 |
| 11 | 1  6e-10 | Округление последнего знака | +0.166666666666666666666666666667E10 |
| 12 | 999999999999999999999999999999  2e0 | Округление с переходом в другие разряды | +0.5E30 |
| 13 | 999999999999999999999999999  999999999999999999999999999e99999 | Предельные значения | +0.1E-99998 |
| 14 | -999999999999999999999999999  -999999999999999999999999999e99999 | Деление минус на минус | +0.1E-99998 |
| 15 | 1  100e99999 | Переполнение | Переполнение (Выход с кодом 3) |
| 16 | 99  -3.3e-99 | Деление с разными знаками | -0.3E97 |
| 17 | 1  10e99999 | Деление | +0.1E-99999 |
| 18 | 10  1e-99999 | Переполнение | Переполнение (Выход с кодом 3) |
| 19 | 12344ф56578666 | Ввод буквы | Ввод не соответствует формату (Выход с кодом 2) |
| 20 | -1  -1e100 | Деление минус на минус | +0.1E-99 |

**Ответы на вопросы**:

1 Каков возможный диапазон чисел, представляемых в ПК?

От 0 до 18 446 744 073 709 551 616 (64 бита без знака)

От -9 223 372 036 854 775 809 до 9 223 372 036 854 775 808 (64 бита со знаком)

2 Какова возможная точность представления чисел? От чего зависит точность вещественного числа.

От -4 503 599 627 370 498 до 4 503 599 627 370 497 для мантисы и от -1025 до 1024 для порядка.

Точность зависит от количества памяти выделенного под мантису.

3 Какие стандартные операции возможны над числами?

сложение, вычитание, умножение, деление.

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

Структуры, включающие в себя: массив цифр, переменную под знак, переменную под порядок.

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

Поразрядным выполнением действий над числом.

**Вывод по лабораторной работе**:

Для хранения и действий с большими числами можно использовать массив. Деление проще всего осуществляется методом в столбик, вещественное число лучше представлять в качестве целой части и порядка.