**Лабораторная работа №1 «Длинная арифметика: обработка больших чисел»**

**Выполнил: Чалый Андрей Александрович**

**Группа: ИУ7-32Б**

**Цель работы:** реализация арифметических операций над числами, выходящими за разрядную сетку персонального компьютера, выбор необходимых типов данных для хранения и обработки указанных чисел.

**Условие задачи:** Смоделировать операцию деления действительного числа 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.

**Исходные данные:**

Вещественное число: число со знаком (или без), состоящее из мантиссы, не превышающей 30 разрядов, и порядка со знаком (или без), не превышающего 5 цифр, со знаком “+”,“-” или без знака (приравнивается к “+”).(2 числа)

**Формат входных данных:**

Данные вводятся в строку.

Пользовательский ввод вещественного числа:

● 12.3E123 , -12.3 E -123 , +12.3 E +123

● 123E12 // без точки

● .123E123

● 0.123E123

● 12.3 // без “E”

**Формат выходных данных:**

Знак 0.мантисса E знак порядок

Пример: +0.12 E -9999

**Возможные ошибки пользователя:**

1. Введенное пользователем вещественное число, мантисса которого больше 30 разрядов.

Текст ошибки: "Ошибка размера ввода мантисcы."

2. Введенное пользователем целое число больше 30 разрядов.

Текст ошибки: "Ошибка размера ввода."

3. Введенное пользователем вещественное или целое число, порядок которого больше 99999 или менее -99999.

Текст ошибки: "Ошибка порядка"

4. Вычисление, приводящее к превышению порядка, не входящего в допустимый предел -99999…99999.

Текст ошибки: "Невозможно произвести вычисление"

6. Некорректный ввод числа.

Текст ошибки: "Ошибка в формате ввода."

**Алгоритм:**

1.Считывание строки вещественного числа

А) Проверка на корректность ввода

Б) Склеивание чисел (если они есть) до точки и после нее

В) Сохранение порядка в отдельную переменную

Г) Приведение строки к нормализованному виду с учетом изменения порядка

2.Считывание строки второго вещественного числа

А) Проверка на корректность ввода

Б) Склеивание чисел (если они есть) до точки и после нее

В) Сохранение порядка в отдельную переменную

Г) Приведение строки к нормализованному виду с учетом изменения порядка

3. Деление чисел, приведенных к нормализованному виду

4. Обработка результата деления

А) Округление

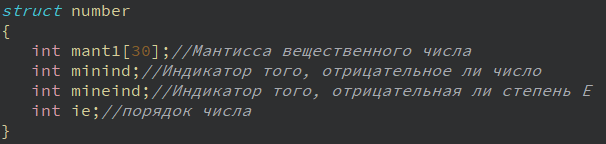
Б) Нахождение порядка

В) Нормализация результата

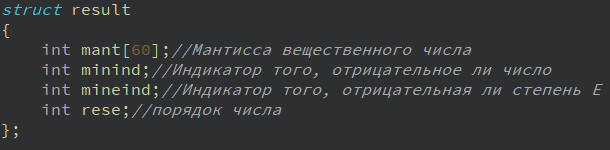
5. Печать результата

**Структуры данных:**

Структура number служит для хранения информации о введенном пользователе целом или вещественном числе



Структура result служит для хранения информации о получившимся при делении числе. Отличие от number в поле mantisa где 60 элементов.



В качестве внутренних структур данных использовались

char num1[100] и char num2[100], которые служили для обработки строк вводимых пользователем данных.

**Функции:**

*//Ввод* *двух* *вещественных* *чисел*

int masinp()

*//Выделение* *новых* *отрезков* *от* *делителя* *для* *последущего* *нахождения* *делителя*

int division(char\* mant1, char\* mant2, int ie1, int ie2, int mantlen2, int mantlen1, int resmantlen1, int minind1, int minind2)

mant1[88], mant2[31] - нормализованное первое и второе вещественное число

ie1, ie2 - значения порядков 1 и 2-го числа

mantlen1, mantlen2 - кол-во цифр в 1 и 2-ом числе

resmantlen1 - кол-во цифр в первом числе до добавления нулей в конец для большей точности

minind1, minind2 - индикатор того отрицательно ли 1 или 2-ое число

*//Нахождение* *делителя* *для* *конкретной* *части* *числа*

int newnum(char\* mant1, char\* mant2, int mantlen2, int mantlen1, int curpos1, int altmas[], int \*altmaslen, int \*lm, int \*prevpos, int \*addtoe, int resmantlen1)

mant1[88], mant2[31] - нормализованное первое и второе вещественное число

mantlen1, mantlen2 - кол-во цифр в 1 и 2-ом числе

curpos1 - крайняя правая позиция текущего отрезка

altmas[62] - массив с результатом деления

altmaslen - длина массива с результатом деления

lm - смещение числа влево если деление было с остатком

prevpos - предыдущая крайняя правая позиция отрезка

addtoe - показатель степени который надо добавить к результату

resmantlen1 - кол-во цифр в первом числе до добавления нулей в конец для большей точности

*//Вывод* *массива*

int masout(int\* altmas, int altmaslen, int e1, int e2, int addtoe, int newe, int minind1, int minind2)

altmas[62] - массив с результатом деления

altmaslen - длина массива с результатом деления

e1, e2 - порядки 1 и 2-го числа

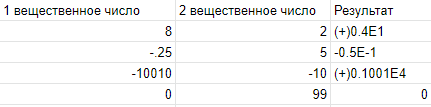
addtoe - добавление к результирующему порядку

newe - порядок вычисленный в результате деления

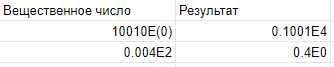
minind1, minind2 - индикатор того отрицательно ли 1 или 2-ое число

**Тесты:**

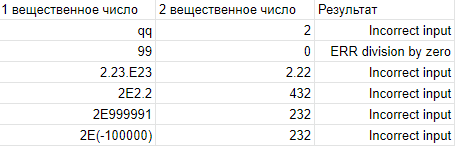
**1. Cтандартный ввод**



**2. Нормализация**



**3.Ввод некорректных данных**



**Контрольные вопросы**

**1. Каков возможный диапазон чисел, представляемых в ПК?**

Диапазон значений чисел зависит от размера области памяти, выделяемой под хранение переменной этого типа, от наличия знака в числе и от типа представления числа (целое или вещественное), от знака (signed). Диапазон чисел, представляемых в ПК, зависит от разрядности процессора. Если процессор имеет 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?**

Точность представления вещественного числа зависит от максимально возможной длины мантиссы. При этом, если мантисса выходит за разрядную сетку ПК, то происходит ее округление. Для 64-разрядного процессора невозможно использовать больше 20 десятичных разрядов для представления целого числа или более 20 знаков после точки в мантиссе для вещественного**.**

**3. Какие стандартные операции возможны над числами?**

Над числами возможно: сложение, вычитание, умножение, деление.

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

Наиболее предпочтительным форматом для хранения числа в памяти ПК является структура, содержащая массив цифр мантиссы, значения показателя и общий знак числа. Данный тип хранения позволяет достаточно быстро и удобно проводить операции над числами.

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

Пользуясь стандартными алгоритмами арифметических операций, таких как умножение и деление в столбик.

**Вывод:**

Таким образом, мною была проведена реализация арифметических операций над числами, выходящими за разрядную сетку персонального компьютера. Я выбрал необходимые типы данных: для хранения я использовал и для обработки указанных чисел я использовал строки. При реализации больших чисел целесообразно использовать массив для хранения цифр числа в строчном или целочисленном виде.