|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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 «Программная инженерия»

**ОТЧЁТ**

**«**Работа № 1 «Длинная» арифметика. Тип данных – массив**»**

Выполнил студент: \_\_***Бугаенко Андрей Павлович***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*фамилия, имя, отчество*

Группа: \_\_\_\_***ИУ7-35Б***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Проверила.**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*подпись, дата*

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*2020 г.*

**Цель работы**

Реализовать арифметические операции над числами, выходящими за разрядную сетку персонального компьютера, выбрать необходимые типы данных для хранения и обработки указанных чисел.

**Условие задания**

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.

Техническое задание

Составить программу деления двух чисел, где порядок имеет до 5 знаков: от –99999 до +99999, а мантисса – до 30 знаков. Программа должна осуществлять ввод чисел и выдавать либо верный результат в указанном формате (при корректных данных), либо сообщение о невозможности произвести счет.

**Исходные данные и результат**:

В качестве исходных данных программа получает два числа, вводимых пользователем с клавиатуры. Формат представления числа - ±m.n Е ±K, где суммарная длина мантиссы (m+n) - до 30 значащих цифр, а величина порядка K - до 5 цифр.

На выходе программа должна вывести в консоль число в форме ±0.m1 Е ±K1, где m1 - до 30 значащих цифр, а K1 - до 5 цифр.

**Описание задачи, реализуемой программой:**

Программа должна считать два числа, которые пользователь вводит с клавиатуры, затем она должна проверить их на корректность и в случае ввода некорректных данных уведомить об этом пользователя. Если же данные были введены корректно, то программа должна представить их в виде соответствующей структуры, и произвести операцию деления. Если во время операции деления возникает ошибка, вызванная некорректностью введённых данных, то программа должна уведомить об этом пользователя. После произведения операции деления программа должна вывести число в форматированном виде.

**Способ обращения к программе:**

Программа запускается из консоли msys, для её запуска вводится команда ./[название приложения].exe. После введения этой команды будет выведен приветственный текст, поясняющий особенности ввода данных. Числа вводятся в программу последовательно. Программа принимает на вход только два числа. После введения первого числа необходимо нажать клавишу «Enter» для того, чтобы число было принято программой. После введения второго числа также нужно нажать «Enter», если числа введены правильно, программа обработает их и выведет на экран результат, иначе будет выведено сообщение об ошибке.

**Описание возможных аварийных ситуаций и ошибок пользователя:**

В случае ошибки во вводе программа прекратит выполнение и выведет на экран текст ошибки, которую допустил пользователь.

Типы ошибок ввода:

* неправильно был введён знак мантиссы - вместо «+», «-», « » или числа был получен любой символ
* неправильно была введена мантисса - вместо символа цифры от 0 до 9 был получен другой символ
* превышена длина мантиссы - длина мантиссы превысила предельно допустимую длину в 30 символов
* неправильно был введён знак экспоненты - вместо символов «е» или «Е» был введён другой символ
* неправильно был введён знак степени - вместо символов «+», «-» или числа был введён другой символ
* неправильно был введена степень - вместо символа цифры от 0 до 9 был получен другой символ

В случае ошибки во вводе программа прекратит выполнение и выведет на экран текст ошибки, которая произошла в ходе деления.

Типы ошибок деления:

* деление на ноль - второе число является нулём в следствие чего деление не может быть совершено

**Структура для хранения больших вещественных чисел:**

В данной программе была реализована структура данных **large\_num**, которая использовалась для представления длинного числа в программе.

Структура содержит следующие поля:

**char mantissa\_sign** - поле для записи знака мантиссы.

**signed short int mantissa\_arr[30]** - массив для записи мантиссы длиной 30 символов. Используется тип signed short int, так как это позволяет сэкономить память и не влияет на производительность программы.

**char power\_sign -** поле для записи знака мантиссы.

**int power** - число для записи числа степени. Массив не используется, так как максимальное число, которое разрешено ввести пользователю, входит в числовой интервал типа int.

**Описание алгоритма:**

Перед началом выполнения основного алгоритма деления необходимо произвести проверки входящих чисел на ноль. Затем рассчитывается начальная степень результата деления.

После этого начинается выполнение алгоритма деления:

Для реализации алгоритма мы используем цикл while, каждый проход мы проверяем числа на успешное деление или на переполнение мантиссы результата.

При проходе через цикл мы сравниваем два числа, если первое больше чем второе, то мы находим множитель для второго, затем отнимает от первого второе, умноженное на делитель. Множитель записывается в последний элемент мантиссы результата.

Если второе больше чем первое, то мы производим либо сдвиг мантиссы первого влево, либо сдвиг мантиссы второго вправо с округлением.

После выхода из цикла while мы проверяем степень на переполнение, а также округляем результат при необходимости.

**Набор тестов:  
Позитивные тесты:**

|  |  |  |  |
| --- | --- | --- | --- |
| первое число | второе число | результат | пояснение |
| 4 | 2 | +0.2 E+1 | Проверка на деление большего на меньшее |
| 1 | 2 | +0.5 E+0 | Проверка на деление меньшего на большее |
| 1 | 3 | +0.333333333333333333333333333333 E+0 | Проверка периодического деления без округления |
| 7 | 9 | +0.777777777777777777777777777778 E+0 | Проверка периодического деления с округлением |
| 6 | 13 | +0.461538461538461538461538461538 E+0 | Проверка деления с округлением |
| 11 | 189 | +0.58201058201058201058201058201 E-1 | Проверка деления с отступами мантиссы |
| +12 | -6 | -0.2 E+1 | Проверка на знак результата |
| -12 | +6 | -0.2 E+1 | Проверка на знак результата |
| -12 | -6 | +0.2 E+1 | Проверка на знак результата |
| .1 | .2 | +0.5 E+0 | Проверка на вариант ввода мантиссы |
| 1.1 E+2 | 2.4 E+1 | +0.458333333333333333333333333333 E+1 | Проверка на вариант ввода мантиссы |
| 0 | 2 | +0.0 E+0 | Проверка на ноль |
| 123456789012345678901234567890 E+99999 | 123456789012345678901234567890 E+99999 | +0.1 E+1 | Проверка на предельное значение |
| 0.1 | 0.2 | +0.5 E+0 | Проверка на вариант ввода мантиссы |
| 1 e+1 | 2 e-2 | +0.5 E+3 | Проверка на вариант ввода степени |
| 2 | 3 | +0.666666666666666666666666666667 E+0 | Проверка на округление |
| 999999999999999999999999999999 | 2 | +0.5 E+30 | Проверка на округление |

**Негативные тесты:**

|  |  |  |
| --- | --- | --- |
| первое число | второе число | пояснение |
| 2 | 999999999999999999999999999999 E+99999 | Проверка на переполнение степени |
| a |  | Проверка на ввод мантиссы |
| +a |  | Проверка на ввод мантиссы |
| -0.a |  | Проверка на ввод мантиссы |
| -12a.123 |  | Проверка на ввод мантиссы |
| -12.a12 |  | Проверка на ввод мантиссы |
| 12 b+10 |  | Проверка на ввод степени |
| 12 E\*12 |  | Проверка на ввод степени |
| 12 E+1a |  | Проверка на ввод степени |
| 9999999999999999999999999999999 |  | Проверка на превышение максимально допустимой длины мантиссы |
| 999999999999999999999999999999 E+100000 |  | Проверка на превышение максимально допустимой длины степени |
| 0.000000000000000000000000000000 E-99999 |  | Проверка на машинный ноль |

**Вывод:**

Использование структуры в данной задаче позволяет реализовать операцию деления для чисел, представление которых является невозможным в рамках стандартных типов, что показывает, насколько структуры могут расширить диапазон данных, используемых для операций вычисления.

**Контрольные вопросы:**

1. Каков возможный диапазон чисел, представляемых в ПК?

Максимальный диапазон целых чисел зависит от количества разрядов, выделяемых для его хранения, если под хранение целого положительного числа выделено 16 разрядов, то его максимальное значение не может превышать 216 -1=65 535, если выделено 32 разряда, то максимальное значение составит 232 -1=4 294 967 295. Для 64 разрядов максимально возможное значение числа равно 264 - 1=18 446 744 073 709 551 615.

Если говорить, о вещественных числах, то максимально длинное число получается при выделении на него 16 байт. В этом случае на знак выделяется один бит, на порядок 15 бит, а на мантиссу 112 бит. При этом возможные значения чисел находятся в диапазоне от 3.6 E –4951 до 1.1 E +4932.

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

Точность представления вещественного числа определяется длиной мантиссы числа. Чем больше разрядов отводится под запись мантиссы, тем выше точность представления числа. В стандартных типах данных на мантиссу максимально отводится 11 разрядов, что не является высокой точностью вычислений.

3. Какие стандартные операции возможны над числами?

Сложение, вычитание, умножение, деление, целочисленное деление, остаток от деления, сохранение знака, отрицание знака, операции отношения.

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

Если числа превышают возможный диапазон представления чисел в ПК, то тогда программисту необходимо выработать собственную структуру данных, которая будет использовать стандартные типы для представления этих чисел.

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

Операции над числами, выходящими за рамки машинного представления, осуществляются аналогично тому, как их выполняет человек. Сложение и вычитание может выполняться в столбик, где размер числа не играет роли. Умножение также выполняется в столбик, так как для этого достаточно реализовать умножение числа на однозначное число, и сложение. Деление можно реализовать с помощью способа «уголка», так как для этого требуется реализация операции умножения многозначного числа на однозначное и операция вычитания двух многозначных чисел.