Отчет по лабораторной работе № 1 по предмету

“Типы и структуры данных”

«Длинная» арифметика. Тип данных – массив

Выполнила:

Кузнецова Анастасия

Группа:

ИУ7-31Б

Вариант 14

Проверил(а):

# **Цель работы:** реализовать арифметические операции над числами, выходящими за разрядную сетку персонального компьютера, выбрать необходимые типы данных для хранения и обработки указанных чисел.

1. **Описание условия задачи**

Смоделировать операцию умножения действительного числа в форме m.n Е 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 0.m1 Е K1, где m1 - до 30 значащих цифр, а K1 - до 5 цифр. Программа должна осуществлять ввод чисел и выдавать либо верный результат в указанном формате (при корректных данных), либо сообщение о невозможности произвести счет.

**2. Описание ТЗ:**

1. Описание исходных данных и результатов

На вход подаются две строки: одна представляет собой действительное число в формате m.n Е K, где суммарная длина мантиссы (m+n) - до 30 значащих цифр, а величина порядка K - до 5 цифр, другая целое число длиной до 30 десятичных цифр.

Оба числа вводятся с клавиатуры и читаются в массив символов.

Ограничения для входных данных: должны быть введены числа в корректной форме, описанной выше.

Результат умножения выводится в виде действительного числа в формате 0.m1 Е K1, где m1 - до 30 значащих цифр, а K1 - до 5 цифр.

1. Описание задачи, реализуемой программой

Необходимо выполнить умножение действительного числа на целое при условии, что числа выходят за разрядную сетку персонального компьютера.

1. Способ обращения к программе

Программу следует вызывать из командной строки с помощью команды:

./a.exe

1. Описание возможных аварийных ситуаций и ошибок пользователей

* Некорректный ввод: нечисловая мантисса или нечисловой порядок в действительном числе (присутствуют недопустимые символы или отсутствуют цифры), нечисловой ввод целого числа (присутствуют недопустимые символы или отсутствуют цифры), суммарная длина мантиссы более 30 значащих цифр или порядок больше 99999 для действительного числа, длина целого числа более 30 десятичных цифр, количество точек в действительном числе больше 1, букв «e» или «E» в действительном числе несколько
* Переполнение порядка в процессе программы

**3. Описание внутренних СД**

Оба введённых числа хранятся в виде статических символьных массивов:

char num1[SIZE\_NUM1 + 1]; //Действительное число SIZE\_NUM1 = 40

char num2[SIZE\_NUM2 + 1]; //Целое число SIZE\_NUM2 = 32

int index\_e; //Индекс символа «e/E»

int index\_point; //Индекс точки

int len1; // Конец действительного числа

int len2; // Конец целого числа

Результат хранится в виде:

int result[RES\_SIZE] = {0}; // статического целочисленного массива результат умножения RES\_SIZE = 60

int exponent; // порядок

int start\_index; // индекс начала числа

**4. Описание алгоритма**

1. Вывод назначения программы
2. Считывание действительного числа в символьный массив
3. Считывание целого числа в символьный массив
4. Нахождение расположения точки и экспоненты в действительном числе
5. Умножение чисел
6. Обновление порядка
7. Округление при необходимости
8. Вывод результата

**5. Набор тестов**

Позитивные тесты:

1. Действительное число введено в формате +m.nE+K (у порядка

знак “+”)

1. Действительное число введено в формате +m.nEK (у порядка нет знака)
2. Действительное число введено в формате +m.nE-K (у порядка

знак “– “)

1. Действительное число введено в формате +m.n (отсутствует порядок)
2. Действительное число введено в формате m.n (отсутствует знак у числа)
3. Действительное число введено в формате -m.n (отрицательное число)
4. Действительное число введено в формате .nE+K (отсутствует целая часть)
5. Действительное число введено в формате E+K (отсутствует мантисса)
6. Действительное число введено в формате mE+K (отсутствует дробная часть)
7. Действительное число введено в формате m (присутствует только целая часть)
8. Умножение 0 на 0
9. Порядок увеличивается
10. Порядок уменьшается
11. В числах 30 цифр
12. Умножение отрицательного на положительное
13. Умножение, после которого требуется округление
14. Округление, приводящее к увеличению порядка
15. Умножение отрицательного на отрицательное
16. Умножение положительного на отрицательное
17. Циклическое округление

|  |  |  |
| --- | --- | --- |
| № | Input | Output |
| 1 | +1.23E+2  30 | +0.369E+4 |
| 2 | +31.0E3  2 | +0.62E+5 |
| 3 | +2.1E-1  2 | +0.42E+0 |
| 4 | +4.5  2 | +0.9E+1 |
| 5 | 9.1  3 | +0.273E+2 |
| 6 | -2.5  4 | -0.1E+2 |
| 7 | .12E+1  4 | +0.48E+1 |
| 8 | E+20  5 | +0.5E+21 |
| 9 | 123E4  1 | +0.123E+7 |
| 10 | 24  4 | +0.96E+2 |
| 11 | 0  0 | +0.0E+0 |
| 12 | 1.2E2  12000 | +0.144E+7 |
| 13 | 0.00005E1  2 | +0.1E-2 |
| 14 | 999999293919294959693949592919  123728463726173694718287378198 | +0.123728376363892793387632363317E+60 |
| 15 | -84.2  50 | -0.421E+4 |
| 16 | 1627892384781984710247  3984792138749182749187 | +0.648681277760891240964000258494E+43 |
| 17 | 100000000000000000000000000001  99999999999999999999999999999 | +0.1E+59 |
| 18 | -2  -6 | +0.12E+2 |
| 19 | 120  -2 | -0.24E+3 |
| 20 | 999999999999999999999999999999  3 | +0.3E+31 |

Негативные тесты:

1. Действительное число имеет больше 1 точки
2. Действительное число имеет больше 1 символа ‘e’/‘E’
3. В действительном числе нечисловая мантисса: присутствуют недопустимые символы
4. В действительном числе без порядка в мантиссе отсутствуют цифры
5. В действительном числе нечисловой порядок: присутствуют недопустимые символы
6. Пустой порядок: в действительном числе после буквы 'e' / 'E' отсутствуют цифры
7. Нечисловой ввод целого числа: присутствуют недопустимые символы
8. Нечисловой ввод целого числа: отсутствуют цифры
9. Суммарная длина мантиссы в действительном числе более 30 значащих цифр
10. Порядок в действительном числе больше 99999
11. Длина целого числа более 30 десятичных цифр
12. Порядок становится больше 99999 при выполнении программы
13. Порядок становится меньше -99999 при выполнении программы
14. Порядок в действительном числе меньше -99999

|  |  |  |
| --- | --- | --- |
| № | Input | Output |
| 1 | 2.5.8  3 | Input error. |
| 2 | 23e4E6  2 | Input error. |
| 3 | 1acd2e1  2 | Input error. |
| 4 | +.  2 | Empty mantissa. |
| 5 | 123e4ab2  3 | Input error. |
| 6 | 2.5E  4 | Empty exponent. |
| 7 | 2.5e1  263hcd | Input error. |
| 8 | 2.1  - | Empty integer. |
| 9 | 1.237364529384765217837465299388  1 | Overflow mantissa. |
| 10 | 12E100000  4 | Overflow exponent. |
| 11 | 3.9  8937485768928374657482736485968 | Overflow integer. |
| 12 | 1.2E99999  4 | Overflow exponent. |
| 13 | 0.001E-99999  1 | Overflow exponent. |
|  | 0.1E-100000  2 | Overflow exponent. |

**6. Выводы по проделанной работе**

В результате выполненной лабораторной работы была написана программа, реализующая умножение действительного числа на целое при условии, что числа мантисса действительного числа имеет до 30 значащих цифр, а в целом числе до 30 десятичных цифр. Таким образом, я научилась выполнять операции над числами, выходящими за разрядную сетку персонального компьютера. Можно сделать вывод, что умножение чисел, выходящих за разрядную сетку можно реализовать с помощью дополнительных структур данных, таких как массив, далее выполнив поразрядное умножение в столбик и получив тем самым верный результат.

**7. Ответы на вопросы**

1. Каков возможный диапазон чисел, представляемых в ПК?

Каждый тип данных характеризуется определенным диапазоном значений чисел, который, в свою очередь, зависит от размера области памяти, выделяемой под хранение переменной этого типа, от наличия знака в числе и от типа представления числа (целое или вещественное). Eсли 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.

Вещественные числа обычно хранятся и используются в представлении с плавающей точкой в виде: X = M \* E^p, где М – мантисса со знаком, Е – основание (10 или 16), р – целый порядок со знаком. Максимально под представление мантиссы отводится 52 разряда, а под представление порядка – 11 разрядов. В этом случае возможные значения чисел находятся в диапазоне от 3.6 E –4951 до 1.1 E +4932.

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

Точность представления вещественного числа зависит от максимально возможной длины мантиссы. При этом, если мантисса выходит за разрядную сетку ПК, то происходит её округление. Максимально под представление мантиссы отводится 52 двоичных разряда, а под представление порядка – 11 разрядов. В этом случае возможные значения чисел находятся в диапазоне от 3.6 E –4951 до 1.1 E +4932.

Для целого числа: для 64-разрядного процессора невозможно использовать больше 20 десятичных разрядов для представления целого числа или более 20 знаков после точки в мантиссе для вещественного.

1. Какие стандартные операции возможны над числами?

Над числами возможно: сложение, вычитание, умножение, деление, остаток от деления, сравнение.

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

Программист сам решает какой тип данных использовать. Наиболее предпочтительным типом для этого является массив, например, массив символов или чисел – для ввода и вывода числа, числовой массив – для обработки. Но, можно использовать структуру для хранения всего числа, например, знак мантиссы, мантисса, знак порядка, порядок. Можно разбить мантиссу на несколько частей, обрабатывая их, а затем «склеивая» число.

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

Можно осуществить стандартные операции над числами, выходящими за рамки машинного представления, с помощью таких структур данных, как массив, структура или разбив мантиссу на несколько частей. Далее поразрядно выполнять операции над числами, например, умножение в столбик или деление в столбик.