**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

**«ОБРАБОТКА БОЛЬШИХ ЧИСЕЛ»**

**ДИСЦИПЛИНА «ТИПЫ И СТРУКТУРЫ ДАННЫХ»**

1. описание условия задачи (вариант 5);

Смоделировать операцию умножения действительного числа на действительное число в форме ±m.n Е ±K, где суммарная длина мантиссы первого сомножителя (m+n) - до 35 значащих цифр, второго – до 40 значащих цифр, а величина порядка K - до 5 цифр. Результат выдать в форме ±0.m1 Е ±K1, где m1 – до 40 значащих цифр, а K1 - до 5 цифр.

2) описание ТЗ, включающем внешнюю спецификацию:

Описание задачи:

Необходимо разработать программу, выполняющую операцию умножения двух действительных чисел в форме ±m.nE±K и возвращающую результат в форме ±0.m1E±K1. Программа должна обрабатывать числа, удовлетворяющие следующим требованиям:

1. Длина мантиссы первого сомножителя (m + n) должна быть не более 35 значащих цифр.

2. Длина мантиссы второго сомножителя должна быть не более 40 значащих цифр.

3. Величина порядка K первого и второго сомножителей должна быть не более 5 цифр.

Входные данные:

Программа должна принимать на вход два действительных числа в формате ±m.nE±K, где m и n - мантисса, K – порядок. Ввод знаков и «Е» обязателен. Между знаком порядка и самим порядком пробел не ставится.

Выходные данные:

Программа должна выводить результат умножения в формате ±0.m1E±K1. Длина мантиссы должна быть не более 40 значащих цифр. Величина порядка K1 должна быть не более 5 цифр, иначе на экран выдаётся ошибка, и вычисления прекращаются.

Обработка ошибок:

В случае возникновения ошибок, таких как переполнение, некорректный формат входных данных или другие недопустимые ситуации, программа должна сообщать об ошибке и запрашивать ввод заново.

b. описание задачи, реализуемой программой:

Цель программы:

Цель программы состоит в выполнении операции умножения двух действительных чисел, представленных в формате ±m.nE±K, и предоставлении результата в формате ±0.m1E±K1. Программа должна обеспечивать высокую точность вычислений и учитывать ограничения на длину мантисс и порядков, а также точность результата.

Описание задачи:

Входные данные:

Программа получает на вход два действительных числа в следующем формате ±m.nE±K, где

- ± знак числа (плюс или минус);

- m и n мантисса первого сомножителя (последовательность цифр);

- K - порядок первого сомножителя (последовательность цифр).

Причем мантисса первого сомножителя (m1 + n1) должна быть до 35 знаков, а мантисса второго сомножителя (m2 + n2) до 40 знаков. Порядок К обоих введённых чисел до 5 цифр.

Выходные данные:

Программа выводит результат умножения в формате ±0.m3E±K3, где:

- ± знак результата (плюс или минус);

- 0.m3 мантисса результата (последовательность цифр);

- K3 порядок результата (последовательность цифр).

Причем мантисса произведения (m3) должна быть до 40 знаков. При превышении этого размера мантисса округляется. Порядок К3 результата должен быть до 5 цифр, иначе на экран выдаётся ошибка, и вычисления прекращаются.

Алгоритм:

Программа должна реализовать алгоритм умножения чисел в формате с плавающей точкой, учитывая следующие особенности:

- Нормализация введённых чисел

- Умножение мантисс;

- Сложение порядков;

- Нормализацию результата;

- Округление;

- Учет знаков.

Обработка ошибок:

Программа должна обрабатывать ошибочные ситуации, такие как:

- Переполнение;

- Некорректный формат входных данных (например, неверное количество цифр или отсутствие знака);

- Другие недопустимые ситуации.

Тестирование

Программа должна пройти тестирование на различных входных данных, включая граничные случаи, чтобы убедиться в правильности выполнения операции умножения и точности результата.

Вывод:

Программа должна реализовать операцию умножения действительных чисел, учитывая формат ±m.nE±K и обеспечивая точность до 40 значащих цифр в результате. Она должна быть готова к обработке ошибок и проходить тестирование на различных входных данных, обеспечивая надежность и точность результатов.

c. способ обращения к программе:

Для запуска программы необходимо в терминале перейти в папку, содержащую исполняемый файл программы, и запустить его командой ./app.exe. Дальнейшее взаимодействие с программой происходит также в терминале.

d. описание возможных аварийных ситуаций и ошибок пользователя.

|  |  |
| --- | --- |
| ***Ошибка пользователя*** | ***Ответ программы*** |
| меню | |
| Некорректный ввод:  На запрос меню ввести 1 или 2 (какое действие выбрать) пользователь ввел буквы. | Вывод сообщения об ошибке и повторный запрос ввода:  Error input. Please try again. |
| Некорректный ввод:  На запрос меню ввести 1 или 2 (какое действие выбрать) пользователь ввел цифру отличную от 1 или 2. |
| Пустой ввод:  На запрос меню ввести 1 или 2 (какое действие выбрать) пользователь нажал Enter. |
| знак | |
| Некорректный ввод:  На запрос ввода знака пользователь ввел другой непробельный символ. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input sign. Please try again. |
| Пустой ввод:  Пользователь не ввёл знак. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input sign. Please try again. |
| мантисса | |
| Некорректный ввод:  Пробельный символ посреди ввода мантиссы. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input mantissa. Please try again. |
| Некорректный ввод:  Слишком длинная мантисса (>35 цифр для первого числа и > 40 знаков для второго. | Вывод сообщения об ошибке и повторный запрос ввода:  Error: mantissa is too long. Please try again. |
| Некорректный ввод:  При вводе мантиссы встречается символ отличный от цифры, пробела, «Е» или точки. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input mantissa. Please try again. |
| Пустой ввод:  Между знаком числа и «Е» нет цифр (отсутствие мантиссы). | Вывод сообщения об ошибке и повторный запрос ввода:  Error: mantissa is empty. Please try again. |
| порядок | |
| Некорректный ввод:  Ошибка при чтении порядка. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input order. Please try again. |
| Некорректный ввод:  Слишком длинный порядок числа (больше 5 цифр). | Вывод сообщения об ошибке и повторный запрос ввода:  Error: order is too long. Please try again. |
| Некорректный ввод:  Пользователь ввел знак порядка и сам порядок через пробел. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input order. Please try again. |
| Результат | |
| Переполнение порядка:  При вычислении порядок превысил 5 цифр. | Вывод сообщения об ошибке и прекращение вычислений:  Program can't calculate this numbers: order overflow. |
|  | |

1. описание внутренних СД:

**//код возврата, если всё хорошо**

**#define OK 0**

**//код возврата, если произошла ошибка**

**#define ERROR 1**

**//масимальная длина мантиссы 1-ого числа**

**#define MAX\_MAN1 35**

**//масимальная длина мантиссы 2-ого числа и результата**

**#define MAX\_MAN23 40**

**//структура - число**

**struct number**

**{**

**int minus; //знак числа**

**int mantissa[MAX\_MAN23 + 2]; //длина мантиссы с запасом два элемента для округления**

**int order; //порядок**

**int len\_man; //длина мантиссы**

**};**

**//первое вводимое число**

**struct number one;**

**//второе вводимое число**

**struct number two;**

**//число - результат их перемножения**

**struct number tree;**

целое число:

**//код ошибки**

**int rc = OK;**

двумерный массив целых чисел:

**int arr[MAX\_MAN23][MAX\_MAN1 + 1] = {{0}};**

символьный тип:

**char a = ' ';**

тип void:

**//функция читает знак числа**

**void input\_sign(struct number \*sth);**

указатели:

**void sum\_diag(struct number one, struct number two, struct number \*tree, int \*h, int \*where, int \*extra, int \*i\_second, int arr[MAX\_MAN23][MAX\_MAN1 + 1]);**

**sum\_diag(one, two, tree, &i\_second, &where, &extra, &h, arr);**

1. описание алгоритма:

При запуске программы на экран выводится приветствие пользователя, условие задачи и меню, предлагающее нажать 1 для начала вычислений или нажать 2 для завершения работы программы.

Если пользователь нажал 1 в меню, программа выводит требования в вводу чисел, подсказывающую разметку и запрос на ввод первого числа. Происходит посимвольное считывание и при ошибке дальнейшая строка не анализируется, выводится сообщение об ошибке и запрашивается повторный ввод. Если ввод первого числа прошёл успешно, выводится подсказывающая разметка для второго числа. Программа анализирует его аналогично первому числу. Данные о сомножителях записываются в структуру (знак числа, массив мантиссы, порядок числа и длина мантиссы).

После успешного ввода чисел запускается модуль вычислений. Создаётся массив 40\*35 и заполняется аналогично алгоритму умножения в столбик: каждая цифра второго числа умножается на все цифры первого числа по очереди, также происходит перенос числа десятков, если при вычислении получается число больше 9. Результат записывается друг под другом. Затем вычисляется место, на которое будут ставиться цифры итоговой мантиссы и при необходимости разряд, с которого нужно начать округление. После этого вычисляется длина итоговой мантиссы и сложением матрицы по диагоналям, параллельным главной, заполняется итоговая мантисса.

Затем вычисляется знак результата и порядок с учетом маски вывода. Если итоговый порядок превысил 5 цифр, выводится сообщение об ошибке и вычисления прекращаются. При необходимости происходит округление со сдвигом числа десятков, если округлённый элемент стал больше 9. После этого выводится результат и снова предлагается выбрать действие в меню.

Если пользователь нажал 2 в меню, программа завершает работу.

1. тесты:

|  |  |  |
| --- | --- | --- |
| Тестовые данные | Что проверяется | Вывод программы |
| 1  +25E2  +11E2  2 | Оба числа со знаком + | Result: +0.275 E +7 |
| 1  -25E2  -11E2  2 | Оба числа со знаком - | Result: +0.275 E +7 |
| 1  -25E2  +11E2  2 | Числа разных знаков | Result: -0.275 E +7 |
| 1  - 25 E 2  + 11 E 2  2 | Ввод с большим количеством пробелов | Result: -0.275 E +7 |
| 2 | Сразу выбор в меню завершения работы | Completion ^-^ |
| 1  + 2E0  + 9999999999999999999999999999999999999999E0  2 | Циклическоеокругление числа | Result: +0.2000000000000000000000000000000000000000 E +41 |
| 1  + 6E0  + 9999999999999999999999999999999999999999E0  2 | Округление последнего разряда числа | Result: +0.5999999999999999999999999999999999999999 E +41 |
| 1  +123456789123456789E-2455  -98765432198765E-99999  2 | Переполнение порядка в меньшую сторону | Program can't calculate this numbers: order overflow. |
| 1  +123456789123456789E-1  -98765432198765E99999  2 | Переполнение порядка в большую сторону | Program can't calculate this numbers: order overflow. |
| 1  -2.5E3  -0E2  2 | Одно из чисел равно нулю | Result: +0.00 E +6 |
| 1  -0E2  -0E2  2 | Оба числа равны нулю | Result: +0.0 E +5 |
| 1  -25E0  -11E0  2 | Порядки нулевые | Result: +0.275 E +3 |
| 1  -25E2  -11E2  1  +6E1  -42E-3  2 | Многократный вызов вычислений в меню | Result: +0.275 E +7  Result: -0.252 E +1 |
| Ferf | Некорректный ввод:  На запрос меню ввести 1 или 2 (какое действие выбрать) пользователь ввел буквы. | Вывод сообщения об ошибке и повторный запрос ввода:  Error input. Please try again. |
| 3 | Некорректный ввод:  На запрос меню ввести 1 или 2 (какое действие выбрать) пользователь ввел цифру отличную от 1 или 2. | Вывод сообщения об ошибке и повторный запрос ввода:  Error input. Please try again. |
| \_ | Пустой ввод:  На запрос меню ввести 1 или 2 (какое действие выбрать) пользователь нажал Enter. | Вывод сообщения об ошибке и повторный запрос ввода:  Error input. Please try again. |
|  | знак |  |
| r324E-32 | Некорректный ввод:  На запрос ввода знака пользователь ввел другой непробельный символ. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input sign. Please try again. |
| 25E2 | Пустой ввод:  Пользователь не ввёл знак. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input sign. Please try again. |
|  | мантисса |  |
| -2 5E2 | Некорректный ввод:  Пробельный символ посреди ввода мантиссы. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input mantissa. Please try again. |
| -33333333333333333333333333333333333333333E2 | Некорректный ввод:  Слишком длинная мантисса (>35 цифр для первого числа и > 40 знаков для второго. | Вывод сообщения об ошибке и повторный запрос ввода:  Error: mantissa is too long. Please try again. |
| -2t5E2 | Некорректный ввод:  При вводе мантиссы встречается символ отличный от цифры, пробела, «Е» или точки. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input mantissa. Please try again. |
| -E2 | Пустой ввод:  Между знаком числа и «Е» нет цифр (отсутствие мантиссы). | Вывод сообщения об ошибке и повторный запрос ввода:  Error: mantissa is empty. Please try again. |
|  | порядок |  |
| -25Et | Некорректный ввод:  Ошибка при чтении порядка. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input order. Please try again. |
| -25E+100000 | Некорректный ввод:  Слишком длинный порядок числа (больше 5 цифр). | Вывод сообщения об ошибке и повторный запрос ввода:  Error: order is too long. Please try again. |
| -\_25E+ 3 | Некорректный ввод:  Пользователь ввел знак порядка и сам порядок через пробел. | Вывод сообщения об ошибке и повторный запрос ввода:  Error with input order. Please try again. |
|  | | |

6) выводы по проделанной работе:

В целом, эта лабораторная работа показала, что при работе с числами большой точности программист может использовать математические алгоритмы работающие с числами поразрядно (умножение/деление/сложение/вычитание столбиком).

Ответы на вопросы:

1. Возможный диапазон чисел, представляемых в ПК:

Диапазон чисел, представляемых в ПК равен 2^64 = 18 446 744 073 709 551 616 (размер шины данных 64).

2. Возможная точность представления чисел и как она определяется:

В памяти компьютера числа с плавающей точкой хранятся: знак (1 бит), порядок (11 бит), и мантисса (52 бита). Мантисса определяет точность хранимого значения. Следовательно, максимальная возможная точность представления чисел 2^52 = 4 503 599 627 370 496.

3. Стандартные операции, возможные над числами:

Стандартные операции включают в себя сложение, вычитание, умножение, деление и сравнение.

4. Тип данных для чисел, превышающих возможный диапазон представления:

Если числа превышают возможный диапазон представления, программист может использовать массивы, структуры, объединения и прочие.

5. Операции над числами, выходящими за рамки машинного представления:

Для операций над числами, выходящими за рамки машинного представления, можно использовать алгоритмы поразрядной обработки.