Отчет по лабораторной работе №1 по типу и структурам данных.

Выполнил: Ильясов Идрис ИУ7-33Б

План:

1) описание условия задачи;

2) ТЗ;

3) описание внутренних структур данных;

4) описанный алгоритм;

5) набор тестов;

6) оценку эффективности;

7) выводы по проделанной работе.

Условие задачи:

Смоделировать операцию умножения 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:

Описание: программа получает на вход два действительных длинных числа, не помещающихся в разрядной сетке ПК, перемножает их и выводит результат.

Заказчик: Силантьева Александра Васильевна.

Основания разработки: программа ВУЗа.

Область применения программы: астрономические вычисления, где нужно оперировать большими числами, необходимы высокоточные вычисления.

Общее описание:

* Рассматриваемая программа представляет собой несколько текстовых файлов, где хранится исходный код, заголовочных текстовых файлов.
* Программа получает на вход два длинных числа, введенных в строго определенном формате, который выводится при запуске программы, перемножает их и выводит результат на экран. При некорректном вводе или переполнении программа выводит соответствующее сообщение на экран.

Требования к программе:

* Программа должна принимать два числа: перемножать их и выдавать результат, если ввод корректный; выводить предупреждение при некорректном вводе или при слишком большом результате.
* При неверных исходных введенных данных, программа завершается, выводя сообщение о причине сбоя и остановки дальнейшей работы.

Внешняя спецификация:

1. Исходные данные:

* Вещественное число записывается в формате ±m.n E ±k, где (m+n) меньше 31 цифры, -99999 ≤ k ≤ 99999
* Допускается опускать n и k, тогда n и k будут равны 0. Также допускается знаки перед числами – в данном случае число «автоматически» будет положительным.

1. Описание аварийных ситуаций: в случае ввода чисел в некорректной форме (если введена слишком большая мантисса, лишние символы, порядок), выводится сообщение об ошибке. В случае переполнения, выводится соответствующее сообщение.

Алгоритм:

1. Считываем вещественное число №1.
2. Если ввод верный, считываем вещественное число №2.
3. Если ввод верный, убираем вещественную часть чисел уменьшением их порядка.
4. Перемножаем два числа
5. Выводим результат.

Функции и их алгоритмы:

**Функция “main()”**

Вывод информации о программе, о том, как вводить число и т.д.;

Считывание первого число;

Считывание второго число;

Удаление вещественной части чисел уменьшением их порядка;

Перемножение двух чисел;

Вывод результата;

**Функция “input()”**

Пока не конец строки:

Считать символ;

Сделать соответствующие символу изменения;

Проверить, вводилось хоть что-то;

**Функция “output()”**

Убрать лишние нули слева от результата;

Проверить порядок на переполнение;

Округлить до 30 значащих цифр;

Убрать лишние нули справа от результата;

Есть целая часть равна нулю:

Вывести 0;

Иначе:

Вывести результат в полной форме;

**Функция “multiply()”**

Обнулить результат;

Перемножая разряды чисел, получить промежуточные слагаемые;

Сложить слагаемые умножения в один результат;

Способ обращения к программе:

Данные вводятся в консоль после запуска исполняемого файла “main.exe”.

Описание структуры данных:

Введенные числа хранятся в следующей структуре:

* char sign\_num\_1, sign\_num\_2 – знак числа №1, числа №2.
* short int num\_1[30], short int num\_2[30] – цифры мантиссы в порядке их следования.
* short int dad\_1, dad\_2 – количество цифр целой части мантиссы чисел №1, №2 до точки.
* short int dbd\_1, dbd\_2 – количество цифр целой части мантиссы чисел №1, №2 после точки.
* int expon\_1, expon\_2 – порядок числа.

Результат умножения хранится следующим образом:

* char sign\_result – знак числа результата.
* short int res[60] – все цифры вещественной части результата в порядке их следования.
* short int len\_res – количество цифр вещественной части результата.
* int expon – порядок результата.

**Тесты**

**(Числа введены корректно)**

Ввод: 2.5

3

Вывод: +0.75 E 1

**(Порядок больше 5 знаков)**

Ввод: 2 E 645123

4

Вывод: Error, wrong input of number #1!

**(Мантисса больше 30 знаков)**

Ввод: 23341235172839201827362712393912313130123

2

Вывод: Error, wrong input of number #2!

**(Корректная обработка знаков)**

Ввод: -2 E 5

3 E 6

Вывод: -0.6 E 12

**(Округление)**

Ввод: 745625499999999999999999999999

3

Вывод: 0.22368765 E 31

**(Переполнение порядка)**

Ввод: 12 E -54532

8 E -86522

Вывод: Overflow!

**(Неверный формат ввода)**

Ввод: 2.84E 2

32

Вывод: Error, wrong input of number #1!

**(Ноль)**

Ввод: 13213 E 100

0 E -3

Вывод: 0

**(Максимальное значение)**

Ввод: +0.99999999999999999999999999999 E 99999

1 E 0

Вывод: +0.99999999999999999999999999999 E 99999

**(“Ложное” переполнение)**

Ввод: 0.0001 E 99999

2

Вывод: 0.2 E 99996

**Вывод по проделанной работе:**

В результате выполнения лабораторной работы я улучшил навыки в работе с длинными вещественными числами, не помещающиеся в разрядной сетке компьютера. Я научился перемножать их, хранить, обрабатывать, вводить и выводить.

**Ответы на контрольные вопросы:**

1. Каков возможный диапазон чисел, представляемых в ПК?

Целые числа:

* short int от 32767 до 32767
* int от -2 147 483 648 до 2 147 483 647
* long int от -9 223 372 036 854 775 807 до 9 223 372 036 854 775 807

Вещественные числа – диапазон зависит от длины мантиссы (чем больше число, тем оно менее точное).

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

Точность представления вещественных чисел зависит от размеров мантиссы. Чем больше число, тем возможность представить его точно меньше, чем если оно будет маленьким. Если число невозможно представить точно, то оно округляется до ближайшего четного числа (16 цифр в мантиссе).

1. Какие стандартные операции возможны над числами?

Возможны следующие стандартные операции над числами: сложение, вычитание, сложение, умножение, операция сравнения.

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

Программист может представить обрабатываемые числа как массивы небольших чисел.

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

Операции над числами, выходящими за рамки машинного представления, можно производить вручную («в столбик» в случае сложения, вычитания, умножения, «уголком» в случае деления).