

Работа № 1

«Длинная» арифметика. Тип данных – массив

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

Краткие теоретические сведения

Современный персональный компьютер (ПК) часто применяется для выполнения вычислений. При этом, для хранения исходных данных и результатов вычислений обычно используются стандартные числовые типы данных (целые и вещественные, в том числе, удвоенной точности). 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).

Целое число X со знаком представляется в ПК следующим образом:

если $X \geq 0$, то число записывается как беззнаковое,

если $X < 0$, то число переводится в дополнительный код и записывается как

$2^k - |X|$, где k – количество разрядов, выделенное под представление числа.

Таким образом, если под хранение целого положительного числа выделено 16 разрядов, то его максимальное значение не может превышать $2^{16}-1=65\,535$, если выделено 32 разряда, то максимальное значение составит $2^{32}-1=4\,294\,967\,295$. Для 64 разрядов максимально возможное значение числа равно $2^{64}-1=18\,446\,744\,073\,709\,551\,615$.

Для 64-разрядного процессора принципиально невозможно использовать больше 20 десятичных разрядов для представления числа, поэтому при необходимости обрабатывать числа большей размерности (например, при выполнении астрономических расчетов) хранение данных и их обработку должен реализовать программист.

Вещественные числа обычно хранятся и используются в представлении с плавающей точкой в виде:

$$X = M * E^p,$$

где M – мантисса со знаком, E – основание (10 или 16), p – целый порядок со знаком.

Если десятичная точка расположена в мантиссе перед первой значащей цифрой числа, то при фиксированном количестве разрядов, отведённых под мантиссу,

обеспечивается возможность сохранить максимальное количество значащих цифр, то есть обеспечить максимальную точность представления числа в ПК. Из сказанного следует, что мантисса должна быть *правильной дробью*, первая цифра которой отлична от нуля, т.е. M находится в интервале $[0.1, 1)$. Такое представление вещественных чисел называется *нормализованным*.

Таким образом, длина мантиссы определяет точность представления числа, а длина порядка ограничивает диапазон допустимых значений. При этом, если мантисса выходит за разрядную сетку ПК, то происходит ее округление. При выходе за заданный диапазон величины порядка могут возникнуть проблемы, связанные с переполнением порядка (при положительном порядке) или получением машинного нуля (при отрицательном порядке).

Максимально под представление мантиссы отводится 52 разряда, а под представление порядка – 11 разрядов. В этом случае возможные значения чисел находятся в диапазоне от $3.6 \text{ E } -4951$ до $1.1 \text{ E } +4932$.

В том случае, если требуется очень высокая точность вычислений (не ниже 20–30 знаков после десятичной точки) или необходимо обрабатывать числа с большим порядком, например превышает 5000) (в навигационных системах или в системах наведения), то задача выбора необходимых структур для хранения и обработки данных и реализации необходимых операций над ними также возлагается на программиста.

Задание

Составить программу умножения или деления двух чисел, порядок которых находится в диапазоне от -99999 до $+99999$ (т.е. имеет не более 5 разрядов), а длина мантиссы не превышает 30 разрядов.

Программа должна осуществлять ввод чисел в указанном диапазоне значений и выдавать результат в нормализованной форме $\pm 0.m1 \text{ E } \pm K1$, где число $m1$ определено до 30 значащих цифр, число $K1$ – до 5 цифр. При невозможности произвести вычисления должно выдаваться соответствующее сообщение.

Указания к выполнению работы

При выполнении лабораторной работы следует обратить внимание, что при хранении чисел в оперативной памяти компьютера необходимо обеспечить следующий формат их представления (рис. 1,2):

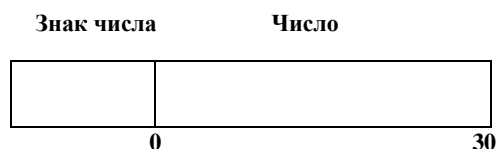


Рис.1. Представление целого числа

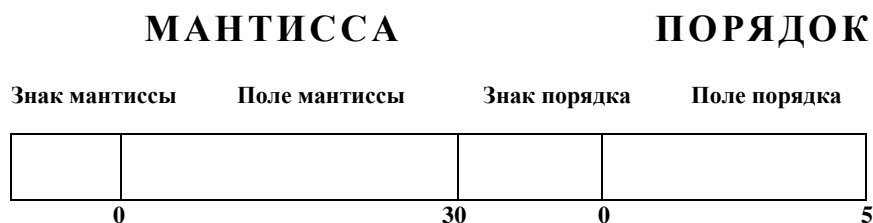


Рис.2. Представление вещественного числа

При наличии десятичной точки в числе возможны следующие варианты его представления: .00025, +123001., –123.456. Также допускается представление числа в экспоненциальной форме: 1234567 E –20, 1234567 E 20 или 123.4567 E23. **В программе должна быть реализована возможность ввода чисел в любом из перечисленных представлений.**

Результат при выдаче на печать должен быть нормализован в виде: знак 0.мантисса E знак порядок.

Указанный формат не имеет стандартного представления в машине, поэтому программист сам должен выбрать типы данных, используя которые возможно осуществить необходимые действия (ввод, вывод, обработку) с данными числами. Наиболее предпочтительным типом для этого является массив, например, массив символов или чисел – для ввода числа, числовой массив – для обработки и вывода.

Проще производить вычисления, если перед обработкой числа нормализованы, т.е. приведены к такому виду, когда после нуля следует значащая цифра. При нормализации порядок входных данных может выйти за пределы указанного диапазона, но при правильных исходных данных, это не должно приводить к ошибке. Для этого необходимо предусмотреть дополнительные разряды в тех структурах данных, которые предназначены для хранения и обработки промежуточных результатов вычисления.

Если при умножении или делении чисел длина мантиссы стала больше 30 знаков, то необходимо произвести округление (если 31-й разряд больше или равен 5, то к 30-му разряду добавляется единица, если меньше 5, то 31-й разряд отбрасывается). При этом может возникнуть циклический поразрядный перенос из младшего разряда в старший с коррекцией порядка. Например, для 5-ти разрядов:

99999 E 01+00008 E 01=100007 E 01 →10001 E 02 .

Так же как нет стандартных типов для хранения таких больших чисел, так нет и стандартных арифметических операций для их обработки, поэтому необходимо разработать эти операции, используя алгоритм умножения и деления в «столбик»

Все логически завершенные фрагменты алгоритма (ввод, вывод, обработка и т.п.) необходимо оформить в виде подпрограмм.

При разработке интерфейса программы следует предусмотреть:

- указание формата и диапазона вводимых данных,
- указание операции, производимой программой,
- наличие пояснений при выводе результата,
- указание формата выводимых данных.

При тестировании программы необходимо:

- проверить правильность ввода и вывода данных (т.е. их соответствие требуемому формату);
- обеспечить вывод сообщений при отсутствии входных данных («пустой ввод»);
- проверить правильность выполнения операций;
- реализовать округление при превышении разрядности мантииссы;
- отследить возникновение переполнения и/или машинного нуля.

Необходимо также протестировать программу на границах допустимых значений данных, задавая самое большое и самое маленькое число в заданном диапазоне представления.

Содержание отчета

Отчет по лабораторной работе должны содержать:

- 1) описание условия задачи;
- 2) описание ТЗ;
- 3) описание СД;
- 4) описание алгоритма (в любом виде);
- 5) набор тестов, с **указанием, что проверяется;**
- 6) выводы по проделанной работе

Кроме того, в отчете должны быть даны ответы на следующие вопросы.

1. Каков возможный диапазон чисел, представляемых в ПК?
2. Какова возможная точность представления чисел, чем она определяется?

3. Какие стандартные операции возможны над числами?
4. Какой тип данных может выбрать программист, если обрабатываемые числа превышают возможный диапазон представления чисел в ПК?
5. Как можно осуществить операции над числами, выходящими за рамки машинного представления?

Отчет представляется в электронном или печатном виде.

Список рекомендуемой литературы

1. *Баженова И.Ю.* Delphi 6. Самоучитель программиста: М.: Кудиц-Образ, 2002. С. 37-38
2. *Савельев А.Я.* Основы информатики: Учеб. для вузов. М.: Изд-во МГТУ им. Н.Э.Баумана, 2001. С. 328.