

УДК:004.451:004.43

Люлюзов М.Ю., Тен И.Г., Мусина И.Р.

КГТУ им. И.Раззакова

ИНФОРМАТИКА: СИСТЕМЫ С ПЛАВАЮЩЕЙ ЗАПЯТОЙ ОГРАНИЧЕННОЙ РАЗРЯДНОСТИ

В данной статье рассматриваются основные причины появления ошибок, неизбежных при любых вычислениях на компьютере. Главное внимание уделено детальному изучению и наглядному представлению арифметических систем ограниченной разрядности/точности, которые используются в любых цифровых компьютерах. Предлагается решение проблемы определения требуемых характеристик процессора для успешной работы программы.

Ключевые слова: компьютерная арифметика, точность вычислений, разрядность процессора, система числа с плавающей запятой.

The article discusses the main causes of the errors that are inevitable in any calculations on a computer. The main attention is paid to the detailed study and a clear presentation systems limited arithmetic word length /precision that is used in all digital computers. It proposed solution to the problem of determining the required processor performance to the success of the program.

Keywords: computer arithmetic, calculation accuracy, bit processor, a floating point number systems.

На сегодняшний день при решении любой задачи, требующей арифметических вычислений на компьютере, пользователь сталкивается с двумя видами ошибок: ошибки входных данных (ошибки действий пользователя) и ошибки вычислений. И если на первый вид ошибок разработчики программ никак повлиять не могут, то ответственность за появление ошибок второго вида полностью возлагается на них. Данная статья посвящена решению проблемы, связанной именно с ошибками вычислений.

Если исключить все ошибки разработчика программы, связанные с неверной реализацией алгоритма, то ошибки в вычислениях могут появляться из-за округления в самой системе, т.к. любая компьютерная система не может быть бесконечно точной. Причиной такого рода ошибок является ограничение памяти и разрядности процессора. К примеру, в обычной арифметике, между нулем и единицей существует бесконечное количество вещественных чисел, а компьютер из-за своей ограниченной разрядности памяти может хранить и использовать только ограниченное количество чисел. В таких случаях компьютеру приходится их округлять. Для представления таких чисел, была введена система чисел с плавающей запятой, что является компромиссом между точностью вычислений и скоростью работы. Число с плавающей запятой состоит из набора отдельных разрядов, условно разделенных на знак, порядок и мантиссу [2]. Порядок и мантисса — целые числа, которые вместе со знаком дают представление числа с плавающей запятой (рис. 1).



Рис.1. Представление числа с плавающей запятой.

Разработка любого программного обеспечения начинается с разработки технического задания (ТЗ). Кроме функциональных требований, в ТЗ должны быть представлены и нефункциональные требования, в которых одним из основных является требования к компьютеру (процессору). В прикладных программах, предназначенных для решения вычислительных задач с большой точностью, в техническое задание на создание прикладной программы должно включаться требование к величине допустимой

погрешности вычислений. Если требования к величине допустимой погрешности завышенные, то это возможно потребует покупки более дорогих компьютеров. Разработчики программ не всегда могут обоснованно предъявить требования к характеристикам аппаратной части компьютера (hardware). Некоторые разработчики, пытаясь сэкономить средства заказчика и сделать более привлекательным стоимость программы, приводят в требованиях к hardware более слабые характеристики, рискуя передать заказчику программу, которая может выдать неудовлетворительный для пользователя программы, результат вычислений. Другие разработчики программного обеспечения завышают требования к аппаратной части компьютерной системы, вынуждая заказчиков приобретать более дорогие компьютеры с высокими характеристиками, в то время, как в этом нет никакой необходимости. Таким образом, возникает необходимость в разработке компьютерной программы, которая будет наглядно демонстрировать зависимость точности вычислений от разрядности процессора.

Цель разработки программы – спрогнозировать/заранее – до покупки компьютера – определить максимально возможную точность для разных процессоров, имеющих различную разрядность. Эта программа позволит программистам в техническом задании на создание прикладной программы правильно сформулировать требования к характеристикам аппаратной части компьютера для достижения требуемой точности вычислений при наименьших затратах на приобретение компьютера.

Для определения максимально возможной точности в программе пользователю будет необходимо лишь ввести некоторые исходные данные, такие как:

- β – Base or radix (основание исчисления). Основание определяет систему счисления разрядов. Математически доказано, что числа с плавающей запятой с базой $\beta = 2$ (двоичное представление) наиболее устойчивы к ошибкам округления, поэтому на практике встречаются только базы 2 и, реже, 10;

- t – Precision (параметр, характеризующий разрядность процессора);
- L – Lower exponent range limit (нижняя граница экспоненциального показателя);
- U – Upper exponent range limit (верхняя граница экспоненциального показателя).

Программа в качестве результатов выдает следующие параметры:

- ✓ Любое число в системе с плавающей запятой представляется следующим образом

[1]:

$$x = \pm \left(d_0 + \frac{d_1}{\beta} + \frac{d_2}{\beta^2} + \dots + \frac{d_{t-1}}{\beta^{t-1}} \right) \beta^e,$$

где

$$0 \leq d_i \leq \beta - 1, \quad i = 0, \dots, t - 1$$

$$L \leq e \leq U.$$

- ✓ Amount of floating-point numbers (Количество всех чисел в системе с плавающей запятой) вычисляется по формуле [1]:

$$2(\beta - 1)\beta^{t-1}(U - L + 1) + 1.$$

- ✓ UFL – Under flow level (Уровень исчезновения разрядов):

$$Underflow\ level = UFL = \beta^L.$$

- ✓ OFL – Over flow level (Уровень переполнения разрядов):

$$Overflow\ level = OFL = \beta^{U+1}(1 - \beta^{-t}).$$

- ✓ ϵ_{mach} – Machine precision (машинная точность). Это – минимальный шаг от одного вещественного числа к другому в заданной системе с плавающей запятой. Перед вычислением значения ϵ_{mach} необходимо определиться со способом округления чисел в системе с плавающей запятой. Существует два основных способа округления:

- Chop – “Рубить” – «отрубают» правую часть вещественного числа, оставляя слева нужное количество символов.

▪ Round to nearest – “Округление до ближайшего”, является режимом по умолчанию и подходит для большинства применений, он обеспечивает наиболее точный результат.

После того, как определились с выбором способа округления, вычисляем значение ϵ_{mach} . Для способа округления Chop, используем формулу [1]:

$$\epsilon_{mach} = \beta^{1-t}.$$

Для способа округления Round to nearest, используем формулу[1]:

$$\epsilon_{mach} = \frac{1}{2}\beta^{1-t}.$$

Выше представленная модель системы с плавающей запятой была реализована в программе, разработанной специально для исследования зависимости точности вычислений от разрядности процессора. Программа разработана на языке C# в среде Visual Studio 2013.

Для исследования зависимости точности вычислений от используемого процессора (процесса округления результатов вычислений) с помощью разработанной программы использовались следующие значения:

Betta (β) = 2 – бинарная система исчисления;

t (параметр, характеризующий разрядность процессора)= 3;

L (нижняя граница экспоненциального показателя) = -1;

U (верхняя граница экспоненциального показателя) = 1;

В качестве способа округления выбран Round to nearest (округление до ближайшего).

После нажатия на кнопку Start программа вычислила:

$$\text{Underflow level} = UFL = 0.5; OFL = 3.5; \epsilon_{mach} = 0.125;$$

$$\text{Amount of floating – point numbers} = 25 .$$

Разработанная нами программа формирует графическое представление системы с плавающей запятой в виде Рисунков 2 и 3, на которых наглядно изображены все точки, возможные при таких исходных данных. Рисунок 2 показывает, что при разрядности процессора равной 3, нельзя получить, например, число 0,25 или 0,3. Если отнять от числа 3 число 0,75, то мы получим 2, т.к. в реализованной в примере системы чисел с плавающей запятой, числа 2,25 не существует.

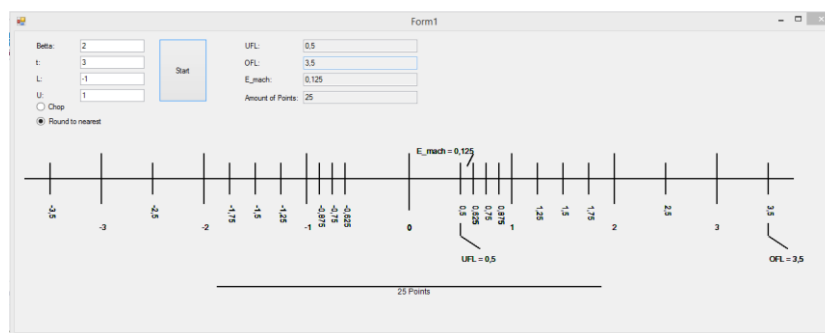


Рис.2. Результаты исследований при t=3.

На рисунке 3 представлены результаты работы программы при увеличении разрядности (t=4). В этом случае, благодаря увеличению точности, мы сможем получить число 2,25, если отнимем 0,75 от числа 3.

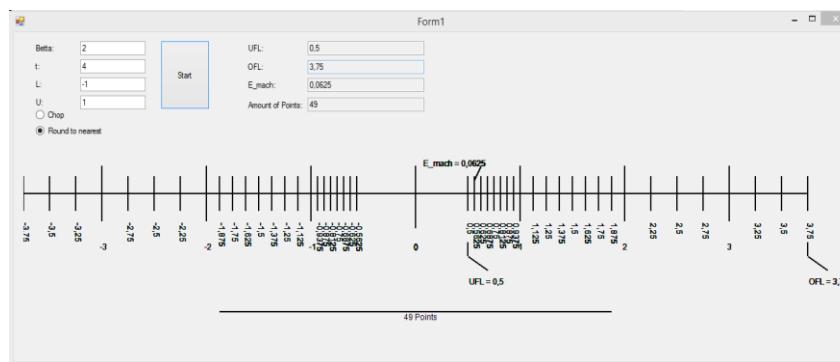


Рис.3. Результаты исследований при $t=4$.

Эти рисунки наглядно показывают, что с увеличением разрядности системы с плавающей запятой увеличивается точность вычисления, а также диапазон значений чисел и их количество.

Были проведены исследования в области компьютерной арифметики, выявлены основные проблемы при формировании технического задания на прикладную программу, предназначенную для решения вычислительных задач. На основе этого анализа была разработана программа, которая наглядно демонстрирует зависимость точности от разрядности процессора. С помощью данной программы можно определить максимально возможную точность для процессоров с различными значениями разрядности процессора. Эта программа позволяет программисту правильно сформулировать требования к допустимой погрешности вычислений при реализации программной системы на уже имеющихся у заказчика типах компьютеров, либо определить характеристики компьютеров, которые надо приобрести, чтобы в техническом задании правильно сформулировать требования к аппаратной части системы для достижения необходимой точности вычислений.

Литература:

1. Heath M.T. Scientific Computing. An Introductory Survey. Boston McGraw Hill, 1997 ISBN 0-07-027684-6.
2. IEEE 754 - стандарт двоичной арифметики с плавающей точкой. <http://www.softelectro.ru/ieee754.html>.