

УДК 004.2

РАЗРАБОТКА ПРОГРАММНОЙ БИБЛИОТЕКИ ДЛЯ РЕАЛИЗАЦИИ ПОСТБИНАРНЫХ ВЫЧИСЛЕНИЙ

А. В. Лях, С. В. Иваница, А. Я. Аноприенко

Донецкий национальный технический университет (г. Донецк)

Рассмотрена проблема современных компьютеров при погрешностях в плавающих вычислениях, а также способы ее решения. Описано создание библиотеки классов с использованием языка моделирования UML. Представлена характеристика бинарных и постбинарных форматов, реализуемых в рассмотренной библиотеке.

В современном мире наблюдается повсеместное использование компьютерных систем, как в бытовых приборах, так и в точных и наукоемких отраслях. Если в первом случае появление ошибки в вычислениях обычно приводит к возникновению незначительных сбоев, то во втором результат может привести к серьезным катастрофам [1]. Поэтому на сегодняшний день вопрос точности компьютерных вычислений в области действительных значений становится особенно актуальным.

Современные компьютеры используют два типа данных: целый тип и тип с плавающей запятой. Целочисленные вычисления являются абсолютно точными, поскольку разрядная сетка целых типов данных полностью покрывает множество целых чисел, определенное разрядностью формата. Для представления вещественных чисел формат состоит из трех полей: знака, порядка и мантиссы. На каждое из полей отводится определенное количество двоичных разрядов в памяти компьютера и чем больше этих разрядов, тем вещественное число представлено более точно. Однако в связи с конечностью двоичных разрядов, отведенных на эти поля, количество возможных чисел тоже конечно и равняется 2^{s+l+m} , где s – разрядность знака, l – разрядность порядка, m – разрядность мантиссы. В свою очередь эти числа представляют конечный набор точек на вещественной оси. Такие точки – точки разрядной сетки формата, не могут отобразить произвольное множество вещественных чисел, являющееся бесконечным на любом диапазоне вещественной числовой оси.

Поэтому при работе с вещественными форматами результаты вычислений часто невозможно задать точками разрядной сетки формата сколь угодно большой точности, не прибегнув к округлению результата. После подобных операций округления теряется точность и, в конечном счете, неизбежно накапливается ошибка вычислений [2].

Одним из решений данной проблемы является использование интервальных вычислений [3], где числа представлены своими крайними значениями с учетом приращения погрешности представления или считывания значения. Однако реализация таких подходов реализуется в настоящее время только программным путем, что резко снижает производительность подобных вычислений. В работе [4] предложено использование постбинарных форматов, что является оптимальным решением, поскольку заложенные в форматах принципы интервального подхода реализуются при постбинарном кодировании вещественных чисел.

Для использования постбинарных форматов при компьютерных вычислениях целесообразна разработка специальной библиотеки. Данная библиотека включает в себя все работу со всеми предложенными в [2] форматами для организации постбинарных вычислений. На сегодняшний день разработаны классы форматов на языке C# с использованием UML-модуля MSVisualStudio [5], в частности диаграмма классов в UML нотации.

Применение UML обусловлено удобством графического представления классов и их наследования, а также проектирования библиотеки. Сокращенная версия диаграммы классов представлена на рис. 4. В дальнейшей реализации для классов представленных форматов будут созданы методы, позволяющие выполнять операции сложения, вычитания, умножения, деления, приведения типов и присваивания.

Данные в представленной библиотеке делятся на 3 типа: числовые (рис. 1), дробные (рис. 2) и интервальные (рис. 3).

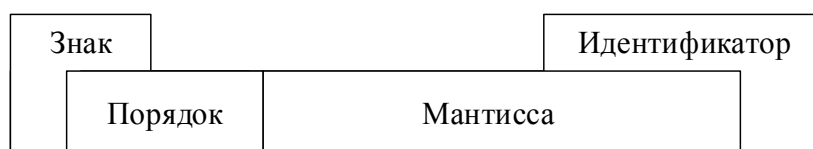


Рис. 1. Числовой тип данных

Каждому типу данных соответствует несколько форматов различной точности, определяемой размером выделяемой памяти для

хранения полей формата. Разрядность данных для каждого типа представлена в табл. 1. Форматы данных BinD32, BinD64, BinD128 и BinD256 – это бинарные форматы данных с плавающей запятой одинарной, двойной, четверной и восьмерной точности соответственно. PBinD32, PBinD64 и PBinD128 – это постбинарные форматы данных соответствующей точности.



Рис. 2. Дробный тип данных

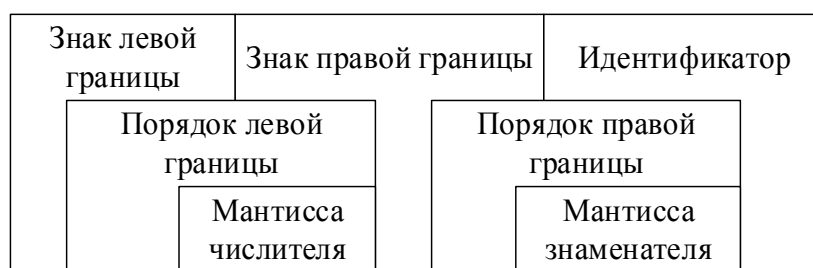


Рис. 3. Интервальный тип данных

Форматы данных BinF32, BinF64 и BinF128 – это бинарные форматы данных дробного типа, в которых в виде чисел с плавающей запятой хранятся числитель и знаменатель. PBinF32, PBinF64 – это постбинарные форматы данных дробного типа одинарной и двойной точности соответственно. Так же, как и в бинарных дробных форматах, в формате хранятся числитель и знаменатель, представленные в постбинарном формате числа с плавающей запятой. Форматы данных BinI32, BinI64 и BinI128 – это бинарные интервальные форматы. В этих форматах хранятся 2 машинных числа, образующие границы интервал, в котором хранится истинное значение, в формате чисел с плавающей запятой одинарной, двойной и четверной точности соответственно. Форматы данных PBinI32 и PBinI64 – это постбинарные интервальные форматы одинарной и двойной точности. Так же, как и в бинарных форматах, они хранят 2 машинных, которые образуют интервал. При реализации методов преобразования десятичных чисел к постбинарным форматам, а также перехода из полей формата к десятичному вещественному числу, разработаны алгоритмы, базирующиеся на работе с целочисленными динамическими значениями по правилам длинной арифметики. В качестве базовой библиотеки была использована GNU Multi-Precision

Libraryfor .NET [6], преимущества которой выражены стабильностью работы, быстрым действием и свободным распространением ПО.

Таблица 1 – Разрядность полей бинарных и постбинарных форматов

Бинарные числовые форматы							
	Знак	Порядок	Мантисса		Идентификатор		
BinD32	1	8	21+1		2		
BinD64	1	10	48+1		4		
BinD128	1	15	104+1		8		
BinD256	1	20	219+1		16		
Постбинарные числовые форматы							
	Знак	Порядок	Мантисса		Идентификатор		
PBinD32	2	16	42+2		4		
PBinD64	2	22	96+2		8		
PBinD128	2	30	208+2		16		
Бинарные дробные форматы							
	Знак дроби	Порядок числителя	Мантисса числителя	Порядок знаменателя	Мантисса знаменателя	Идентификатор	
BinF32	1	8	21+1	8	21+1	4	
BinF64	1	11	48+1	11	48+1	8	
BinF128	1	15	104+1	15	104+1	16	
Постбинарные дробные форматы							
	Знак дроби	Порядок числителя	Мантисса числителя	Порядок знаменателя	Мантисса знаменателя	Идентификатор	
PBinF32	2	16	42+2	16	42+2	8	
PBinF64	2	22	96+2	22	96+2	16	
Бинарные интервальные форматы							
	Знак левой границы	Порядок левой границы	Мантисса левой границы	Знак правой границы	Порядок правой границы	Мантисса правой границы	Идентификатор
BinI32	1	8	21+1	1	8	21+1	4
BinI64	1	11	48+1	1	11	48+1	8
BinI128	1	15	104+1	1	15	104+1	16
Постбинарные интервальные форматы							
	Знак левой границы	Порядок левой границы	Мантисса левой границы	Знак правой границы	Порядок правой границы	Мантисса правой границы	Идентификатор
PBinI32	2	16	42+2	2	16	42+2	8
PBinI64	2	22	96+2	2	22	96+2	16

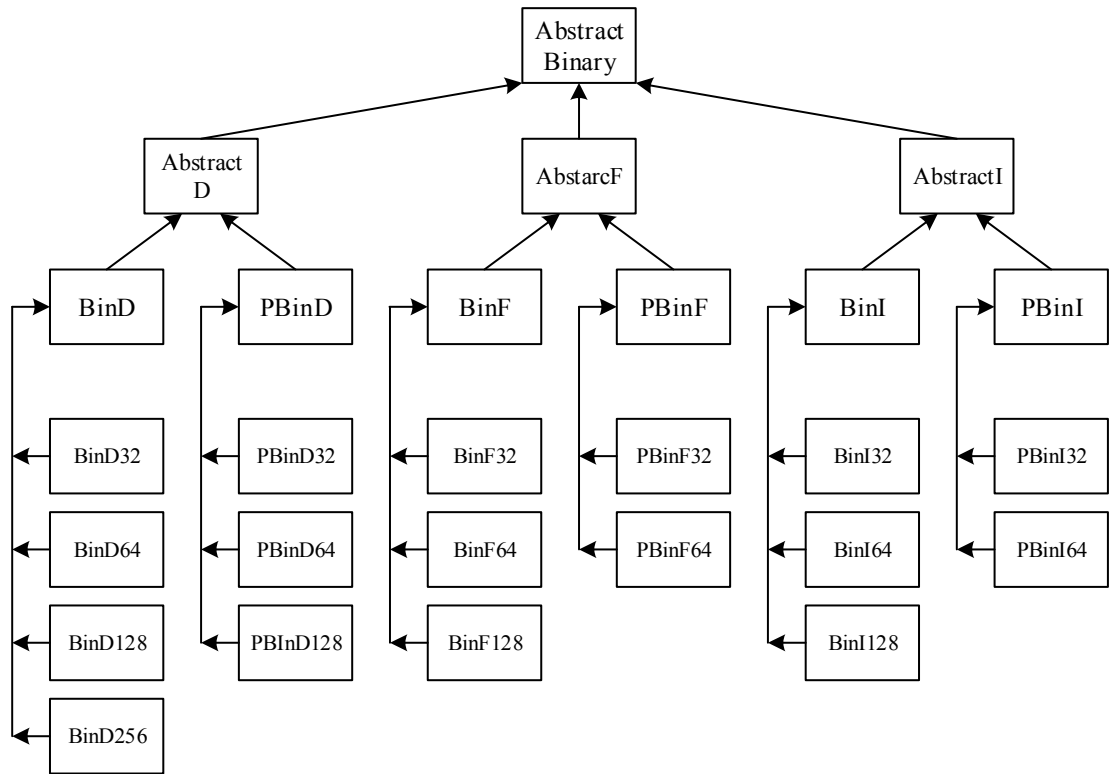


Рис. 4. Сокращенная версия диаграммы классов

Список литературы:

1. Петров Ю. П. Обеспечение достоверности и надежности компьютерных расчетов. / Ю. П. Петров — СПб.: БХВ-Петербург, 2012. — 160 с.: ил.
2. Аноприенко А.Я., Иваница С.В. Постбинарный компьютинг и интервальные вычисления в контексте кодо-логической эволюции. — Донецк, ДонНТУ, УНИТЕХ, 2011. — 248 с. Режим доступа: <http://ea.donntu.org:8080/jspui/bitstream/123456789/19453/3/2012-anoprienko-ivanitsa-tetracomputing-308-p.pdf>.
3. Шарый С.П. Конечномерный интервальный анализ. — Новосибирск, Институт вычислительных технологий СО РАН, 2009. — 569 с.
4. Аноприенко А. Я., Иваница С. В. Тетралогика, тетравычисления и ноокомпьютинг. Исследования 2010–2012. — Донецк: ДонНТУ, Технопарк ДонНТУ УНИТЕХ, 2012. — 308 с. Режим доступа: <http://ea.donntu.org:8080/jspui/bitstream/123456789/19453/3/2012-anoprienko-ivanitsa-tetracomputing-308-p.pdf>.
5. Архитектурные инструменты в VisualStudio 2010. Электронный ресурс. — Режим доступа: <http://habrahabr.ru/post/68334>.
6. The GNU Multiple Precision Arithmetic Library. Электронный ресурс. — Режим доступа: <https://gmplib.org/>.