КОНСТРУИРОВАНИЕ ПРОГРАММ

Лекция № 02

Системы счисления и представление чисел в ЭВМ

Преподаватель: Поденок Леонид Петрович, 505а-5

+375 17 293 8039 (505а-5)

+375 17 320 7402 (ОИПИ НАНБ)

prep@lsi.bas-net.by

ftp://student:2ok*uK2@Rwox@lsi.bas-net.by/

Кафедра ЭВМ, 2022

2022.02.18

Оглавление

Системы счисления и представление чисел в ЭВМ

Математическое понятие числа

- натуральные – $\mathbb{N}$ (natural);
- целые – $\mathbb{Z}$ (integer, integral);
- рациональные – $\mathbb{Q}$ (rational);
- действительные (вещественные) – $\mathbb{R}$ (real);
- комплексные – $\mathbb{C}$ (complex);
- гиперкомплексные (кватернионы) $\mathbb{H}$ (quaternion, hypercomplex).

Свойства числовых множеств
- замкнутость;
- коммутативность (перестановочность);
- ассоциативность (сочетательность);
- дистрибутивность (распределительность);
- существование обратного числа;
- существование противоположного числа.

Свойство замкнутости некоторого множества относительно математической операции означает, что результат операции принадлежит этому множеству

$$c = a \star b \quad a \in M, b \in M, c \in M.$$

Свойство коммутативности бинарной операции $\star$ (переместительности)

$$a \star b = b \star a.$$

Пример: $a + b = b + a; \quad a \cdot b = b \cdot a;$

Свойство ассоциативности бинарной операции $\star$ (сочетательности)

$$(a \star b) \star c = a \star (b \star c).$$

Пример:
$$a + b + c = (a + b) + c = a + (b + c),$$
$$a \cdot b \cdot c = (a \cdot b) \cdot c = a \cdot (b \cdot c).$$

Свойство дистрибутивности бинарной операции $\circ$ (распределительности) относительно бинарной операции $\star$

$$a \circ (b \star c) = (a \circ b) \star (a \circ c).$$

Пример: $a \cdot (b + c) = (a \cdot b) + (a \cdot c).$

Обратное и противоположное числа

$$b \cdot a = 1,$$
$$b + a = 0.$$

Множество целых чисел $\mathbb{Z}$

Целые числа — числа, получаемые объединением натуральных чисел с множеством чисел противоположных натуральным и нулём

$$\mathbb{Z} = \{\cdots -2, -1, 0, 1, 2, \dots\}.$$

Противоположное число

$$b + a = 0.$$

Множество натуральных чисел входит в множество целых чисел

$$\mathbb{N} \subset \mathbb{Z}.$$

Любое целое число можно представить как разность двух натуральных.

Целые числа замкнуты относительно сложения, вычитания и умножения (но не деления); в общей алгебре такая алгебраическая структура называется кольцом (ring).

Вычеты по модулю $\mathbb{Z}/n$

Если два целых числа a и b при делении на целое m дают одинаковые остатки, то они называются сравнимыми (или равноостаточными) по модулю числа m. Записывается так:

$$a \equiv b \pmod{m}$$

Число m называется модулем.

Любое целое число при делении на m дает один из m возможных остатков – от 0 до $m-1$. Это значит, что все целые числа можно разделить на m групп, каждая из которых отвечает определённому остатку от деления на m. Эти остатки называются вычетами по модулю m.

Арифметические операции с остатками чисел по фиксированному модулю образуют модульную (модулярную) арифметику, которая широко применяется в математике, информатике и криптографии.

Вычеты по модулю простого числа $\mathbb{Z}/p$ замкнуты также относительно деления — в общей алгебре такая алгебраическая структура называется конечным полем или полем Галуа (finite field или Galois field).

$$7 + 5 = 1 \mod 11;$$
$$\langle 7 + 5 \rangle_{11} = 1;$$
$$\langle 7 \cdot 5 \rangle_{11} = 2;$$
$$\langle 7 \div 5 \rangle_{11} = 8;$$

Проверим: $5 \cdot 8 = 40 \equiv 7 \pmod{11}$

`https://ru.wikipedia.org/wiki/Сравнение_по_модулю`

Виноградов И. М. Основы теории чисел. — М.—Л.: Гос. изд. технико-теоретической литературы, 1952. — С. 41—45. — 180 с.

Множество рациональных чисел $\mathbb{Q}$

Рациональные числа — числа, представимые в виде дроби

$$\frac{m}{n}, \quad (n \neq 0).$$

где m – целое число, а n — натуральное. Иногда n полагают натуральным числом, возлагая ответственность за знак дроби на числитель.

Формальное определение:

$$\mathbb{Q} = \left\{ \frac{m}{n} \mid m \in \mathbb{Z},\ n \in \mathbb{N} \right\}. \tag{1}$$

Еще одно формальное определение – множество классов эквивалентности пар:

$$\{(m,\ n) \mid m, n \in \mathbb{Z}, n \neq 0\} \tag{2}$$

Рациональные числа замкнуты относительно всех четырёх арифметических действий — сложения, вычитания, умножения и деления (кроме деления на ноль).

В общей алгебре такая алгебраическая структура называется полем (field).

Обратное число

$$b \cdot a = 1,$$

Свойства:

- сложение, умножение и деление;
- коммуативность сложения и умножения;
- ассоциативность сложения и умножения;
- дистрибутивность умножения относительно сложения;
- наличие нуля и единицы;
- наличие противоположных чисел;
- наличие обратных чисел;
- множество рациональных чисел **счётно** (все рациональные числа можно пронумеровать, т.е. установить биекцию между множествами рациональных и натуральных чисел).
- в *позиционной системе счисления* рациональное число представляется *периодической дробью*.

Наличие представления в виде периодической дроби является критерием рациональности вещественного числа.

$$(1/2)_2 = 0,10000\ldots = 0,1\{0\}$$
$$(1/2)_{10} = 0,50000\ldots = 0,5\{0\}$$
$$(1/3)_2 = 0,010101\ldots = 0,\{01\}$$
$$(1/3)_{10} = 0,33333\ldots = 1,\{3\}$$

Между любыми двумя различными рациональными числами a и b существует бесконечно много рациональных чисел. Иначе говоря, не существует двух соседних рациональных чисел. В частности, не существует наименьшего положительного рационального числа.

Множество действительных чисел $\mathbb{R}$

Действительные (вещественные) числа — числа, представляющие собой расширение множества рациональных чисел $\mathbb{Q}$, замкнутое относительно некоторых важных для математического анализа операций (*квадратура круга, соизмеримость*...).

Множество действительных чисел $\mathbb{R}$ включает множество рациональных чисел $\mathbb{Q}$ и множество иррациональных чисел $\mathbb{I}$, не представимых в виде отношения целых.

Действительные числа подразделяются на алгебраические и трансцендентные.

При этом каждое действительное трансцендентное является иррациональным, а каждое рациональное число — действительным алгебраическим.

Алгебраическое число — корень многочлена (не равного тождественно нулю) с коэффициентами из $\mathbb{Q}$

$$0 = P^n(x) = a_0 + a_1\,x + a_2\,x^2 + \cdots + a_n\,x^n \quad a_k \in \mathbb{F}. \tag{3}$$

Поле алгебраических чисел обычно обозначается $\mathbb{A}$.

Трансцендентное число — это вещественное или комплексное число, не являющееся алгебраическим — иными словами, число, которое не может быть корнем многочлена с рациональными коэффициентами (не равного тождественно нулю).

Примеры трансцендентных чисел: число π; число e; десятичный логарифм любого натурального числа, кроме чисел вида $10^{\pm n}$; $\sin a$, $\cos a$ и $\operatorname{tg} a$ для любого ненулевого алгебраического числа a.

https://ru.wikipedia.org/wiki/Вещественное_число

Множество комплексных чисел $\mathbb{C}$

Комплексные числа — числа, являющиеся расширением множества действительных чисел $\mathbb{R}$. Они могут быть записаны в виде

$$z = x + i\,y,$$

где i — т. н. «мнимая» единица, для которой выполняется равенство $i^2 = 1$.

Комплексные числа используются при решении задач электротехники, гидродинамики, картографии, квантовой механики, теории упругости и многих других.

Комплексные числа подразделяются на алгебраические и трансцендентные, как и действительные.

Множество гиперкомплексных чисел $\mathbb{H}$

Гиперкомплексные числа — числа, являющиеся расширением множества комплексных чисел $\mathbb{C}$. Они могут быть записаны в виде

$$\lambda = a + i\,b + j\,c + k\,d,$$

где a, b, c и d — действительные числа; i, j и k — «мнимые» единицы, для которых выполняются следующие соотношения

$$i^2 = j^2 = k^2 = -1, \quad i \cdot j = k = -j \cdot i, \quad j \cdot k = i = -k \cdot j, \quad k \cdot i = j = -i \cdot k.$$

Гиперкомплексные числа (кватеринионы) широко используются в теоретической механике для представления поворотов, а также в инерциальной навигации.

изированное оборудование, например регистры перемещения. Их значение прибавляется к каждому адресу, сгенерированному процессом. Например, x86 использует четыре таких (сегментных) регистра.Представление чисел

Для перечисленных множеств чисел справедливо следующее выражение

$$\mathbb{N} \subset \mathbb{Z} \subset \mathbb{Q} \subset \mathbb{R} \subset \mathbb{C} \subset \mathbb{H}.$$

При этом каждое множество из указанных справа от символа «$\subset$» может быть сконструировано из множеств, указанных слева.

Система счисления (numeral system/system of numeration) — символический метод записи чисел, представление чисел с помощью письменных знаков, например букв или цифр.

Система счисления должна:

- дать каждому числу уникальное (или, по крайней мере, стандартное) представление;
- отражать алгебраическую и арифметическую структуру чисел.

Системы счисления подразделяются на:

- позиционные;
- непозиционные;
- смешанные.

Позиционная система счисления

Позиционная система счисления — система счисления, в которой значение каждого числового знака (цифры) в записи числа зависит от его позиции (разряда).

Позиционная система счисления определяется целым числом $b > 1$, называемым основанием системы счисления.

Система счисления с основанием b также называется b-ичной (двоичной, троичной, восьмеричной, десятичной и т.п.).

Целое число без знака x (натуральное с нулем) в b-ичной системе счисления представляется в виде конечной линейной комбинации степеней числа b:

$$x = a_0 + a_1\, b + a_2\, b^2 + \cdots + a_{n-2}\, b^{n-2} + a_{n-1}\, b^{n-1} = \sum_{k=0}^{k} a_k\, b^k \tag{4}$$

где a_k — целые числа, называемые ***цифрами***, удовлетворяющими неравенству $0 \leqslant a_k < b$.

Данное представление является единственным.

$$x = a_0 + a_1\, 10 + a_2\, 10^2 + \cdots + a_{n-2}\, 10^{n-2} + a_{n-1}\, 10^{n-1} = \sum_{k=0}^{k} a_k\, 10^k \tag{5}$$

Каждый базисный элемент b^k таком представлении называется разрядом (позицией).

Старшинство разрядов и соответствующих им цифр определяется номером разряда k, который является показателем степени. С помощью n позиций в системе счисления с основанием b можно записать целые числа в диапазоне от 0 до $b^n - 1$, т.е. всего b^n различных чисел.

Непозиционные системы счисления

В непозиционных системах счисления величина, которую обозначает цифра, не зависит от положения в числе.

Система остаточных классов (СОК). Residue number system (RNS)

Представление числа в системе остаточных классов основано на понятии вычета и китайской теореме об остатках.

Вычет числа n по модулю p – остаток от деления n на p

$$a = k \cdot p + r, \quad r < p.$$

Записывается, как

$r = a \mod p = \langle a \rangle_p$ – математическая запись

```
r = a % p; -- в языке программирования Си.
```

СОК определяется набором попарно взаимно простых модулей $(m_1, m_2, \ldots, m_n)$ с произведением $M = m_1 \cdot m_2 \cdot \cdots \cdot m_n$ так, что каждому целому числу x из отрезка $[0, M-1]$ ставится в соответствие набор вычетов $(x_1, x_2, \ldots, x_n)$, где $x_k = \langle x \rangle_{m_k}$.

При этом **китайская теорема об остатках** гарантирует однозначность такого представления для чисел из отрезка $[0, M-1]$.

Преимущества:

В СОК арифметические операции (сложение, вычитание, умножение, деление) выполняются покомпонентно, если про результат известно, что он является целочисленным и также лежит в диапазоне $[0, M-1]$.

Недостатки:

- возможность представления только ограниченного количества чисел;
- отсутствие эффективных алгоритмов для сравнения чисел, представленных в СОК;
- вычислительная сложность деления (нахождения обратной величины).

Используется для выполнения операций с большими целыми числами, в частности в криптографии и для точного решения плохо обусловленных линейных систем высокого порядка.

Двоичная система счисления

Двоичная система счисления — позиционная система счисления с основанием 2. Непосредственно реализуется в цифровых электронных схемах на логических вентилях, в связи с чем используется практически во всех современных компьютерах и прочих вычислительных электронных устройствах.

Двоичная запись чисел

В двоичной СС числа записываются с помощью пары символов (0 и 1). Обычно записанное число снабжают указателем справа внизу. Например, число в десятичной системе 5_{10}, в двоичной 101_2.

Иногда двоичное число обозначают префиксом **`0b`**, например **`0b101`**.

Натуральные числа

Натуральное число, записываемое в двоичной системе счисления как

$$(a_{n-1}a_{n-2}\ldots a_1a_0)_2,$$

имеет значение:

$$(a_{n-1}a_{n-2}\ldots a_1a_0)_2 = \sum_{k=0}^{n-1} a_k 2^k, \tag{6}$$

где: n — количество цифр (знаков) в числе, a_k — цифры из множества $\{0, 1\}$, k — порядковый номер цифры.

Отрицательные числа

Отрицательные двоичные числа обозначаются так же как и десятичные — знаком «−» перед числом. Отрицательное целое число, записываемое в двоичной системе счисления $(-a_{n-1}a_{n-2}\ldots a_1a_0)_2$, имеет величину:

$$(-a_{n-1}a_{n-2}\ldots a_1a_0)_2 = -\sum_{k=0}^{n-1} a_k 2^k.$$

В вычислительной технике широко используется запись отрицательных двоичных чисел в **дополнительном коде**.

```
0100 — +4
0011 — +3
0010 — +2
0001 — +1
0000 —  0              Инверсия
1111 — -1      -0001     →     1110 + 1 → 1111      | → 0000 +1 → 0001 (1)
1110 — -2      -0010     →     1101 + 1 → 1110      | → 0001 +1 → 0010 (2)
1101 — -3      -0011     →     1100 + 1 → 1101      | → 0010 +1 → 0011 (3)
1100 — -4      -0100     →     1011 + 1 → 1100      | → 0011 +1 → 0100 (4)
       -8      -1000     →     0111 + 1 = 1000 = 8


-1    1111...1111 1111
```

Дробные числа

Дробное число можно представить, используя отрицательные степени основания

$$x = a_{n-1}\, b^{n-1} + a_{n-2}\, b^{n-2} + \cdots + a_2\, b^2 + a_1\, b + a_0 + a_{-1}\frac{1}{b} + a_{-2}\frac{1}{b^2} + \cdots \qquad (7)$$

и отделять в записи дробную часть от целой с помощью запятой. Тогда число, записываемое в двоичной системе счисления как $(a_{n-1}\, a_{n-2} \ldots a_1\, a_0\ ,\ a_{-1}\, a_{-2} \ldots a_{-(m-1)}\, a_{-m})_2$, будет иметь величину:

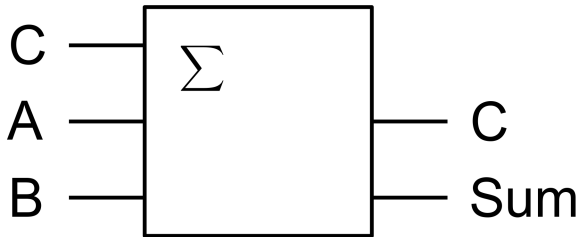
$$(a_{n-1}a_{n-2} \ldots a_1 a_0\ ,\ a_{-1}a_{-2} \ldots a_{-(m-1)}a_{-m})_2 = \sum_{k=-m}^{n-1} a_k 2^k,$$

где: m – число цифр дробной части числа, a_k – цифры из множества $\{0, 1\}$.

Сложение чисел, записанных в двоичной СС

Таблица сложения (наивная)

	0	1
0	0	1
1	1	10



	C	7 (128)	6 (64)	5 (32)	4 (16)	3 (8)	2 (4)	1 (2)	0 (0)
Перенос C (Carry)		1	1	1	0	0	0	0	?
Слагаемое A ($1101\ 0010_2 = 210_{10}$)		1	1	0	1	0	0	1	0
Слагаемое B ($1011\ 1001_2 = 185_{10}$)		1	0	1	1	1	0	0	1
Сумма S		1	0	0	0	1	0	1	1
Перенос C	1	1	1	1	1	0	0	0	0

Сложение чисел, записанных в двоичной СС

Таблица сложения

C	A	B	$\sum$	C	Sum
0	0	0	0	0	0
0	0	1	1	0	1
0	1	0	1	0	1
0	1	1	2	1	0
1	0	0	1	0	1
1	0	1	2	1	0
1	1	0	2	1	0
1	1	1	3	1	1

Умножение чисел, записанных в двоичной СС

Таблица умножения

	0	1
0	0	0
1	0	1

	C	7 (128)	6 (64)	5 (32)	4 (16)	3 (8)	2 (4)	1 (2)	0 (0)
Сомножитель A ($1101_2 = 13_{10}$)						**1**	**1**	**0**	**1**
Сомножитель B ($1010_2 = 10_{10}$)						**1**	**0**	**1**	**0**
						0	0	0	0
					1	**1**	**0**	**1**	
				0	0	0	0		
			1	**1**	**0**	**1**			
Произведение P		**1**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
Перенос C				1	1	1			

Вычитание. Представление отрицательных чисел в дополнительном коде

Вычитание можно представить так

$$c = a - b = a + (-b).$$

Число 210:	1	1	0	1	0	0	1	0

1) наивное представление — добавляем знаковый разряд

S	1	1	0	1	0	0	1	0

s = 0 — число положительное 210, s = 1 — число отрицательное -210.

Вычитание, однако, не упрощается — необходимо иметь разную аппаратуру для сложения положительных отрицательных чисел. Плюс, у нас появилось два нуля.

2) Число в дополнительном коде

Исходное число	210	1	1	0	1	0	0	1	0

Но отрицательное число формируем следующим образом:

Инвертируем все разряды	45	0	0	1	0	1	1	0	1
Добавляем единицу		0	0	0	0	0	0	0	1
Число в дополнительном коде	46	0	0	1	0	1	1	1	0
Проверяем	(210 + 46) % 256 = 0	0	0	0	0	0	0	0	0

Операции над двоичными представлениями [целых чисел]

Кроме целых чисел упорядоченные последовательности 0 и 1 используются для двоичного представления и других информационных объектов, например, битовых строк, символов некоторого алфавита, символьных строк и прочих.

Операции над числами и прочими двоичными представлениями делятся на арифметические и логические.

Они бывают одноместные (унарные) и двуместные (бинарные).

Одноместные:

- логические — поразрядная инверсия 0010 1101 → 1101 0010; (not)
- арифметические — смена знака 0010 1101 → 1101 0011; (neg)
- сдвиги.

Двуместные:

- арифметические — сложение, вычитание, умножение, целочисленное деление.
- логические — поразрядные AND, OR, XOR.

AND

	0	1
0	0	0
1	0	1

OR

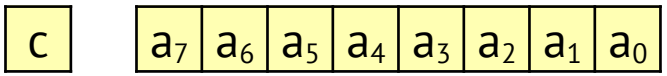
	0	1
0	0	1
1	1	1

XOR

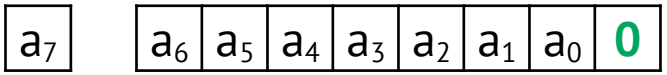
	0	1
0	0	1
1	1	0

	0	1
0	1	0
1	0	1

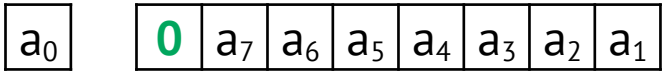
Сдвиги



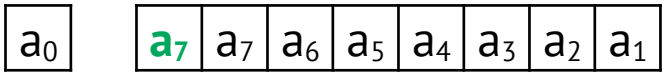
x



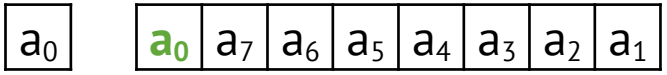
влево (логический/арифметический) sal, sll



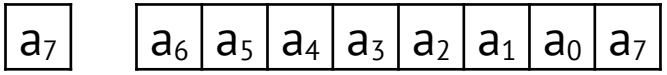
вправо логический slr



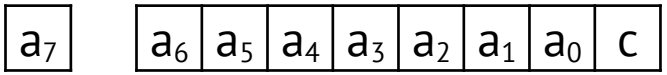
вправо арифметический sar



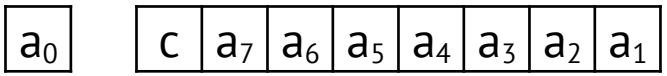
вправо циклический ror



влево циклический rol



влево циклический через перенос rcr



вправо циклический через перенос rcl

s — shift
r — roll

a — arithmetic
l — logic
c — carry

l — left
r — right

Другие типы данных

Кроме чисел есть другие типы данных, которые могут быть представлены в двоичном виде.

В области телекоммуникаций и компьютерных технологий используется стандарт, описывающий структуры данных для представления, кодирования, передачи и декодирования данных — **ASN.1** (Abstract Syntax Notation One).

Начиная с 1995 года, существенно пересмотренный ASN.1 описывается стандартом X.680.

В России ASN.1 стандартизирован по:

ГОСТ Р ИСО/МЭК 8824-1-2001

ГОСТ Р ИСО/МЭК 8825-93

Особенности вычислений с плавающей запятой в двоичном представлении

1) разные компьютеры — разные способы представления, соответственно, разные результаты при выполнении вычислений. Например, вычисления на CPU и GPU дадут разные результаты;

2) неассоциативность — результат вычислений зависит от порядка выполнения действий;

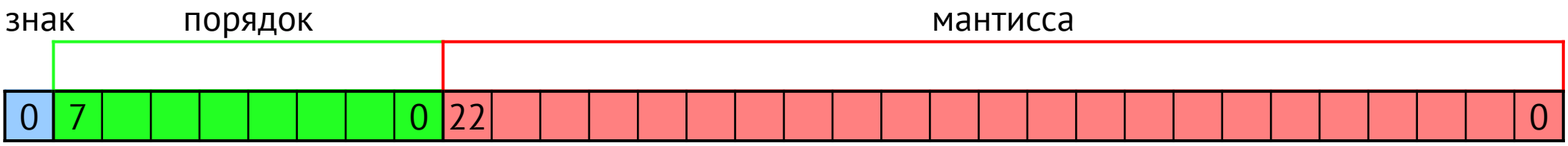
3) некоммутативность — результат вычислений зависит от перестановки операндов местами;

4) ноль имеет знак;

5) разность неравных чисел дает ноль.

Формат числа с плавающей запятой (плавающей точкой)

Число с плавающей запятой (или число с плавающей точкой) — форма представления действительных чисел, в которой число хранится в форме **мантиссы** и **показателя степени**.



При этом число с плавающей запятой имеет **фиксированную относительную точность** и **изменяющуюся абсолютную**.

Математически это записывается следующим образом:

$$(-1)^s \times M \times B^E,$$

где s — знак числа, M — мантисса, B — основание (2 или 10), E — порядок.

Мантисса — это целое число фиксированной длины, которое представляет старшие разряды действительного числа.

Порядок — это степень основания (базы) старшего разряда.

Примеры «научного» формата — 1.01e+2, 1.00110011e-16

Математически доказано, что числа с плавающей запятой с основанием 2 (двоичное представление) наиболее устойчивы к ошибкам округления, поэтому на практике встречаются только основания 2 и, реже, 10.

Реализация математических операций с числами с плавающей запятой в вычислительных системах может быть как аппаратная, так и программная.

Используемое наиболее часто представление утверждено в стандарте **IEEE 754**.

Нормальная форма и нормализованная форма

Нормальная форма числа с плавающей запятой называется такая форма, в которой мантисса (без учёта знака) находится на полуинтервале [0; 1) ($0 \leq a < 1$).

Нормальная форма записи имеет недостаток: некоторые числа записываются неоднозначно (например, 0,0001 можно записать в 4-х формах:

$$0,0001 \cdot 10^0; \quad 0,001 \cdot 10^1; \quad 0,01 \cdot 10^2; \quad 0,1 \cdot 10^3$$

и это очень неудобно с точки зрения аппаратной реализации вычислителя.

Нормализованная форма

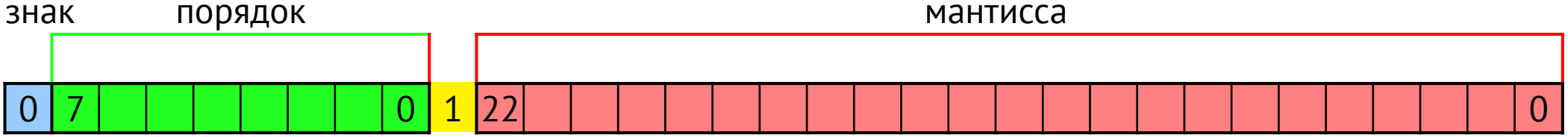
мантисса десятичного числа принимает значения в диапазоне [1; 10) ($1 \leqslant a < 10$)

мантисса двоичного числа принимает значения в диапазоне [1; 2) ($1 \leqslant a < 2$).

В нормализованной форме любое число кроме 0 записывается единственным образом.

Недостаток: в таком виде невозможно представить 0, поэтому компьютерное представление чисел с плавающей запятой предусматривает специальный формат представления нуля.

Так как старший разряд (целая часть числа) мантиссы двоичного числа (кроме 0) в нормализованном виде всегда равен «1», то при записи мантиссы числа в ЭВМ старший разряд можно не записывать, что и используется в стандарте **IEEE 754**. Это экономит один бит, так как неявную единицу не нужно хранить в памяти, и обеспечивает уникальность представления числа.



В позиционных системах счисления с основанием большим, чем 2 (в троичной, четверичной и др.), этого свойства нет.

Стандарт IEEE 754

IEEE 754 (Ай трипл И)— технический стандарт формата представления чисел с плавающей точкой, используемый как в программных реализациях арифметических действий, так и во многих аппаратных (CPU и FPU) реализациях.

Формат представляет собой «набор представлений числовых величин и символов», а также включает правила, согласно которым этот набор кодируется.

Изначально принят в 1985 как IEEE 754-85 институтом IEEE.

В настоящее время используется IEEE 754-2019, который является небольшой ревизией стандарта IEEE 754-2008, заменившего IEEE 754-1985 и включающего оригинальный IEEE 754-85 и стандарт IEEE 854-1987 (IEEE Standard for Radix-Independent Floating-Point Arithmetic). Имеется европский стандарт ISO/IEC/IEEE 60559:2011 (идентичный американскому IEEE 754).

В стандарт IEEE 754-2008 включены все бинарные форматы из первоначального стандарта, а также три новых базовых формата. В соответствии с действующим стандартом, реализация должна выполнять обработку по крайней мере одного из основных форматов арифметики и одного из форматов обмена.

Многие компиляторы языков программирования используют этот стандарт для хранения данных и выполнения математических операций.

Стандарт описывает

Определенные числа (finite number) по основанию 2 и 10. Каждое определенное число (как представлено в стандарте) описывается следующими целыми:

s — знак (0 или 1);

p — точность (precision);

c — мантисса (significand или коэффициент), имеющая не более p цифр по основанию b (т.е. представляющая целое от 0 до $b^p - 1$;

q — показатель (exponent), лежащий в диапазоне $E_{min} \leqslant q + p - 1 \leqslant E_{max}$.

и представляется как

$$(-1)^s \times c \times b^{q-p+1},$$

где b — основание (*base* или *radix*), равное 2 или 10.

Эффективное значение порядка сдвинуто и равно q — 127 для чисел одинарной точности (`float`).

Кроме того, существует два нулевых значения, которые называются *нулями со знаком*. Знаковый бит s определяет является ли ноль +0 (положительный ноль) или -0 (отрицательный ноль).

Две бесконечности : $+\infty$ и $-\infty$.

+INF s = 0, экспонента = 1, мантисса = 0;

-INF s = 1, экспонента = 1, мантисса = 0

Денормализованные числа

все биты экспоненты равны 0.

Неопределенность:

s = 1;

все биты экспоненты = 1;

первый бит мантиссы (2 для 80 битного числа) = 1;

остальные 0.

Два типа NaN:

qNaN — тихий, мягкий (quiet) — попав в операцию, возвратит NaN;

sNaN — сигнализирующий (signaling) — попав в операцию, вызовет исключение;

Формат NaN:

32-bit NaN: s111 1111 1axx xxxx xxxx xxxx xxxx xxxx

s — знак (в приложениях игнорируется);

x — дополнительная полезная информация (в приложениях игнорируется).

a = 1 — мягкий NaN;

a = 0 и доп. информация не равна нулю — сигнализирующий NaN.

Также стандарт описывает:

Методы, которые используются для преобразования числа в процессе математических операций;

Обработку исключительных ситуаций, таких как деление на нуль, переполнение, потеря значимости, работу с денормализованными числами и т.д.

Операции — арифметические и другие операции по арифметике форматов.

Name	Обычное название	Base	Digits	E min	E max	Decimal digits	Decimal E max
binary16	половинная точность	2	11	−14	+15	3.31	4.51
binary32	одинарная точность	2	24	−126	+127	7.22	38.23
binary64	двойная точность	2	53	−1022	+1023	15.95	307.95
binary128	четырехкрат. точность	2	113	−16382	+16383	34.02	4931.77
binary256	восмикратная точность	2	237	−262142	+262143	71.34	78913.2
decimal32		10	7	−95	+96	7	96
decimal64		10	16	−383	+384	16	384
decimal128		10	34	−6143	+6144	34	6144

Десятичное E max - это $emax \times \log_{10} основание$ —(максимальная степень в десятичном формате)

Машинная эпсилон

В отличие от чисел с фиксированной запятой, сетка чисел, которые способна отобразить арифметика с плавающей запятой, неравномерна — она более густая для чисел с малыми порядками и более редкая — для чисел с большими порядками (Логарифмическая равномерность).

Относительная погрешность записи чисел одинакова и для малых чисел, и для больших. Поэтому можно ввести понятие машинной эпсилон.

Машинной эпсилон называется наименьшее положительное число ε такое, что $1 \oplus \varepsilon \neq 1$ (знаком $\oplus$ обозначено машинное сложение).

Числа a и b, соотносящиеся так, что $1 < \frac{a}{b} < 1 + \varepsilon$, машина не различает.

В стандарте C/C++ ISO

```
#include <cmath> (math.h)
#include <limits> // определяет темплейт класса

std::numeric_limits<double>::epsilon() (gcc ≈ 1 · 10^−16)

#include <cfloat> (float.h)
FLT_EPSILON, DBL_EPSILON, LDBL_EPSILON
```

Свойства чисел и их компьютерных представлений

Сложение и умножение натуральных, целых, рациональных и вещественных чисел подчиняются следующим алгебраическим законам:

1) свойство замкнутости

Результат математической операции над элементами множества принадлежит этому же множеству.

2) коммутативный (переместительный) закон;

$$a + b = b + a \qquad a \cdot b = b \cdot a$$

3) ассоциативный (сочетательный) закон;

$$(a + b) + c = a + (b + c) \qquad (a \cdot b) \cdot c = a \cdot (b \cdot c)$$

4) дистрибутивный (распределительный) закон;

$$a \cdot (b + c) = a \cdot b + a \cdot c$$

Для машинных представлений вещественных чисел свойство замкнутости, ассоциативный и дистрибутивный законы в общем случае не выполняются из-за ограниченной разрядной сетки.