

ТСИС

(Технические средства информационных систем)

Программное обеспечение информационных систем (1-40 01 73)

Гр. 6 0 3 2 5, 6 0 3 2 6

Формат с плавающей запятой. Стандарт IEEE
754. Погрешности. Обратная польская запись

Лекция 2

(По материалам Мухаметова В.Н.)

Ковалевский Вячеслав Викторович

Ковалевский Вячеслав Викторович

4096tb@gmail.com

Тема письма:
БГУИР.



Лекция 1. Представление информации. Системы счисления.

Формат с фиксированной запятой

План лекции:

- История развития вычислительной техники.
- Понятие информации.
- Принцип программного управления.
- Двоичная и шестнадцатеричная системы счисления.
- Прямой и дополнительный код.
- Арифметические действия в Формате ФЗ.
- Переполнение.

Экзаменационные вопросы:

- Информационная система. Информация. История развития компьютера.
- Позиционные системы счисления. Перевод чисел из одной системы счисления в другую.
- Арифметика ЭВМ. Представление чисел в форме с фиксированной точкой.
- Сложение в формате с фиксированной точкой. Переполнение.
- Операция вычитания с фиксированной точкой. Дополнительный код числа.

Лекция 2. Формат с плавающей запятой. Стандарт IEEE 754. Погрешности. Обратная польская запись

План лекции:

- Формат чисел с плавающей запятой.
- Стандарт IEEE 754.
- Особенности операций в формате с плавающей запятой.
- Переполнение порядков.
- Точность вычислений.
- Обратная польская запись.

Экзаменационные вопросы:

- Представление чисел в форме с плавающей точкой. Мантисса и характеристика числа.
- Нормализованные и денормализованные числа. Погрешность представления числа.
- Арифметические операции в формате с плавающей точкой.
- Стандарт IEEE 754.
- Формат BCD. Представление текстовой информации. ASCII.

Двоично-десятичный код (BCD)

(англ. *binary-coded decimal*), BCD, 8421-BCD

— это форма записи рациональных чисел, при которой каждый десятичный разряд числа записывается в виде его четырёхбитного двоичного кода.

Двоично-десятичный код (BCD)

Десятичное число:

311_{10}

В двоичной системе счисления в двоичном коде:

$1\ 0011\ 0111_2$

В двоично-десятичном коде:

$0011\ 0001\ 0001_{\text{BCD}}$

Преимущества BCD

- Упрощён вывод чисел на индикацию — вместо последовательного деления на 10 требуется просто вывести на индикацию каждый полубайт. Проще ввод данных с цифровой клавиатуры.
- Для дробных чисел (как с фиксированной, так и с плавающей запятой) при переводе в человекочитаемый десятичный формат и наоборот не теряется точность.
- Упрощены умножение и деление на 10, а также округление.
- Двоично-десятичный формат удобен в калькуляторах — результат, равен подсчитываемому человеком на бумаге.

Двоично-десятичный код (BCD)

При помощи 4 бит можно закодировать 16 цифр. Из них используются 10. Остальные 6 комбинаций в двоично-десятичном коде являются запрещенными.

Двоично-десятичный код				Десятичный код
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9

BCD также применяется в телефонной связи. Для записи символов '*' или '#' и т.п. в двоично-десятичном коде используются запрещенные комбинации:

Двоично-десятичный код				Дополнительный символ
1	0	1	0	* (звёздочка)
1	0	1	1	# (решётка)
1	1	0	0	+ (плюс)
1	1	0	1	- (минус)
1	1	1	0	, (десятичная запятая)
1	1	1	1	Символ гашения

Сложение и вычитание чисел (BCD)

$$\begin{array}{r} + 11 \\ + 15 \\ \hline 26 \end{array}$$

$$\begin{array}{r} 0001 \ 0001 \\ + 0001 \ 0101 \\ \hline \underbrace{0010}_2 \ \underbrace{0110}_6 \end{array}$$

$$\begin{array}{r} - 18 \\ - 11 \\ \hline 7 \end{array}$$

$$\begin{array}{r} \ (011) \ \text{заем} \\ - 0001 \ 1000 \\ - 0001 \ 0001 \\ \hline \underbrace{0000}_0 \ \underbrace{0111}_7 \end{array}$$

Сложение и вычитание чисел (BCD)

$$\begin{array}{r}
 +15 \\
 +16 \\
 \hline
 31
 \end{array}
 \quad
 \begin{array}{r}
 +0001\ 0101 \\
 +0001\ 0110 \\
 \hline
 0010\ \underline{1011}
 \end{array}$$

межтетрадный
перенос
перенос

$$\begin{array}{r}
 +18 \\
 +18 \\
 \hline
 36
 \end{array}
 \quad
 \begin{array}{r}
 +0001\ 1000 \\
 +0001\ 1000 \\
 \hline
 0011\ 0000
 \end{array}$$

межтетрадный заем

$$\begin{array}{r}
 -15 \\
 -9 \\
 \hline
 6
 \end{array}
 \quad
 \begin{array}{r}
 (0) \swarrow \downarrow \\
 0001\ 0101 \\
 -0000\ 1001 \\
 \hline
 0000\ 1100
 \end{array}$$

заем

Сложение и вычитание чисел (BCD)

$$\begin{array}{r}
 15 \\
 + 16 \\
 \hline
 31
 \end{array}$$

$$\begin{array}{r}
 0001\ 0101 \\
 + 0001\ 0110 \\
 \hline
 0010\ 1011 \\
 + 0110 \\
 \hline
 0011\ 0001
 \end{array}$$

запрещенная тетрада

← коррекция

3 1

$$\begin{array}{r}
 18 \\
 + 18 \\
 \hline
 36
 \end{array}$$

$$\begin{array}{r}
 0001\ 1000 \\
 + 0001\ 1000 \\
 \hline
 0011\ 0000 \\
 + 0110 \\
 \hline
 0011\ 0110
 \end{array}$$

межтетрадный перенос

← коррекция

3 6

$$\begin{array}{r}
 15 \\
 - 9 \\
 \hline
 6
 \end{array}$$

$$\begin{array}{r}
 0001\ 0101 \\
 - 0000\ 1001 \\
 \hline
 0000\ 1100 \\
 - 0110 \\
 \hline
 0000\ 0110
 \end{array}$$

межтетрадный заем

(0) заем

← коррекция

0 6

Сложение и вычитание чисел (BCD)

$$\begin{array}{r}
 + 999 \\
 + \quad 6 \\
 \hline
 1005
 \end{array}$$

$$\begin{array}{r}
 + 1001 \ 1001 \ 1001 \\
 + 0000 \ 0000 \ 0110 \\
 \hline
 + 1001 \ 1001 \ (1111) \\
 0110 \quad \leftarrow \text{первая коррекция} \\
 \hline
 + 1001 \ (1010) \ 0101 \\
 0110 \quad \leftarrow \text{вторая коррекция} \\
 \hline
 + 1010 \ 0000 \ 0101 \\
 0110 \quad \leftarrow \text{третья коррекция} \\
 \hline
 \underbrace{0001}_1 \ \underbrace{0000}_0 \ \underbrace{0000}_0 \ \underbrace{0101}_5
 \end{array}$$

Обратная польская запись

Форма записи математических и логических выражений, в которой операнды расположены перед знаками операций.

Также именуется как обратная польская нотация (ОПН), обратная бесскобочная запись (ОБЗ), постфиксная нотация, бесскобочная символика Лукасевича, польская инверсная запись, ПОЛИЗ.

Алгоритм, проводящий вычисления по обратной польской записи называется стековой машиной.

Обратная польская запись



$$\perp \quad A * (B + C) \quad \perp$$

Вагон перед стрелкой

Вагон в Техасе (стек)

	⊥	+	P	*	/	()
⊥	4	1	1	1	1	1	5
+	2	2	2	1	1	1	2
P	2	2	2	1	1	1	2
*	2	2	2	2	2	1	2
/	2	2	2	2	2	1	2
(5	1	1	1	1	1	3

Текущая запись

Последняя запись в
стеке

	\perp	$+$	P	$*$	$/$	$($	$)$
\perp	4	1	1	1	1	1	5
$+$	2	2	2	1	1	1	2
P	2	2	2	1	1	1	2
$*$	2	2	2	2	2	1	2
$/$	2	2	2	2	2	1	2
$($	5	1	1	1	1	1	3

Числа соответствуют
следующим ситуациям:

1. Вагон на стрелке отправляется в Техас
2. Последний вагон, направившийся в Техас, разворачивается и направляется в Калифорнию
3. Вагон, находящийся на стрелке, и последний вагон, отправившийся в Техас, угоняются и исчезают

4. Остановка. Символы, находящиеся на Калифорнийской ветке, представляют собой формулу в обратной польской записи, если читать слева направо
5. Остановка. Произошла ошибка. Изначальная формула была некорректно сбалансирована

В ОПЗ: $(8+2*5)/(1+3*2-4)$

Шаг	Стрелка	Стек (Техас)	Результат (Калифорния)	Оставшаяся цепочка
0	0	0	0	$\perp(8+2*5)/(1+3*2-4) \perp$
1	\perp	\perp	0	$(8+2*5)/(1+3*2-4) \perp$
2	(\perp (0	$8+2*5)/(1+3*2-4) \perp$
3	8	\perp (8	$+2*5)/(1+3*2-4) \perp$
4	+	\perp (+	8	$2*5)/(1+3*2-4) \perp$
5	2	\perp (+	82	$*5)/(1+3*2-4) \perp$
6	*	\perp (+*	82	$5)/(1+3*2-4) \perp$
7	5	\perp (+*	825	$)/(1+3*2-4) \perp$
8)	\perp (+	825*	$)/(1+3*2-4) \perp$
9)	\perp (825*+	$)/(1+3*2-4) \perp$
10)	\perp	825*+	$)/(1+3*2-4) \perp$
11	/	\perp /	825*+	$(1+3*2-4) \perp$
12	(\perp /(825*+	$1+3*2-4) \perp$

Шаг	Стрелка	Стек (Техас)	Результат (Калифорния)	Оставшаяся цепочка
13	1	\perp /(825*+1	$+3*2-4) \perp$
14	+	\perp /(+	825*+1	$3*2-4) \perp$
15	3	\perp /(+	825*+13	$*2-4) \perp$
16	*	\perp /(+*	825*+13	$2-4) \perp$
17	2	\perp /(+*	825*+132	$-4) \perp$
18	-	\perp /(+	825*+132*	$4) \perp$
19	-	\perp /(825*+132*+	$4) \perp$
20	-	\perp /(-	825*+132*+	$4) \perp$
21	4	\perp /(-	825*+132*+4	$) \perp$
22)	\perp /(825*+132*+4-	\perp
23)	\perp /	825*+132*+4-	\perp
24	\perp	\perp	825*+132*+4-/	0
25	\perp		0825*+132*+4-/	0

$$= 825*+132*+4-/$$

Действия в ОПЗ: $825^*+132^*+4- /$

Шаг	Оставшаяся цепочка	Стек
1	8, 2, 5, $*+1, 3, 2^*+4- /$	8
2	2, 5, $*+1, 3, 2^*+4- /$	8, 2
3	5 $*+1, 3, 2^*+4- /$	8, 2, 5
4	$*+1, 3, 2^*+4- /$	8, 10
5	$+1, 3, 2^*+4- /$	18
6	1, 3, 2 $*+4- /$	18, 1
7	3, 2 $*+4- /$	18, 1, 3
8	2 $*+4- /$	18, 1, 3, 2
9	$*+4- /$	18, 1, 6
10	$+4- /$	18, 7
11	4 $- /$	18, 7, 4
12	$- /$	18, 3
13	$/$	6

Таблица ASCII

ASCII (от англ. American Standard Code for Information Interchange) — американский стандартный код для обмена информацией. ASCII представлена в виде таблицы печатных символов и некоторых специальных управляющих символов, каждому символу соответствует уникальный код в диапазоне от [0;255].

ASCII представляет собой кодировку для представления десятичных цифр, латиницы и кириллицы, знаков препинания и управляющих символов

Таблица ASCII

ASCII Code Chart

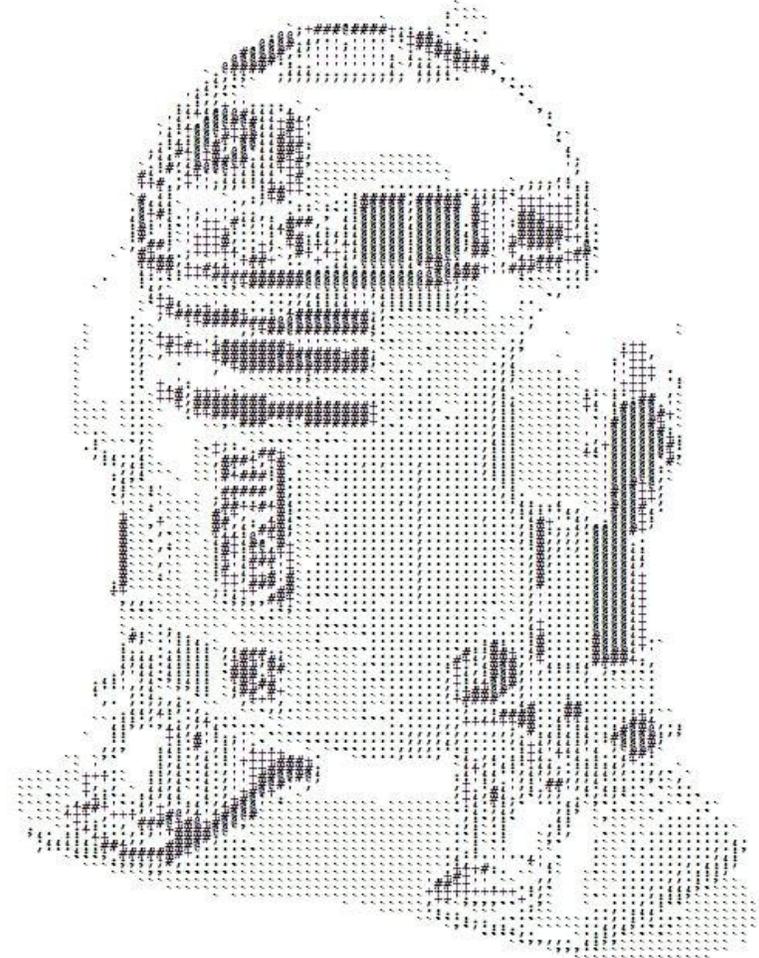
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

ASCII-графика



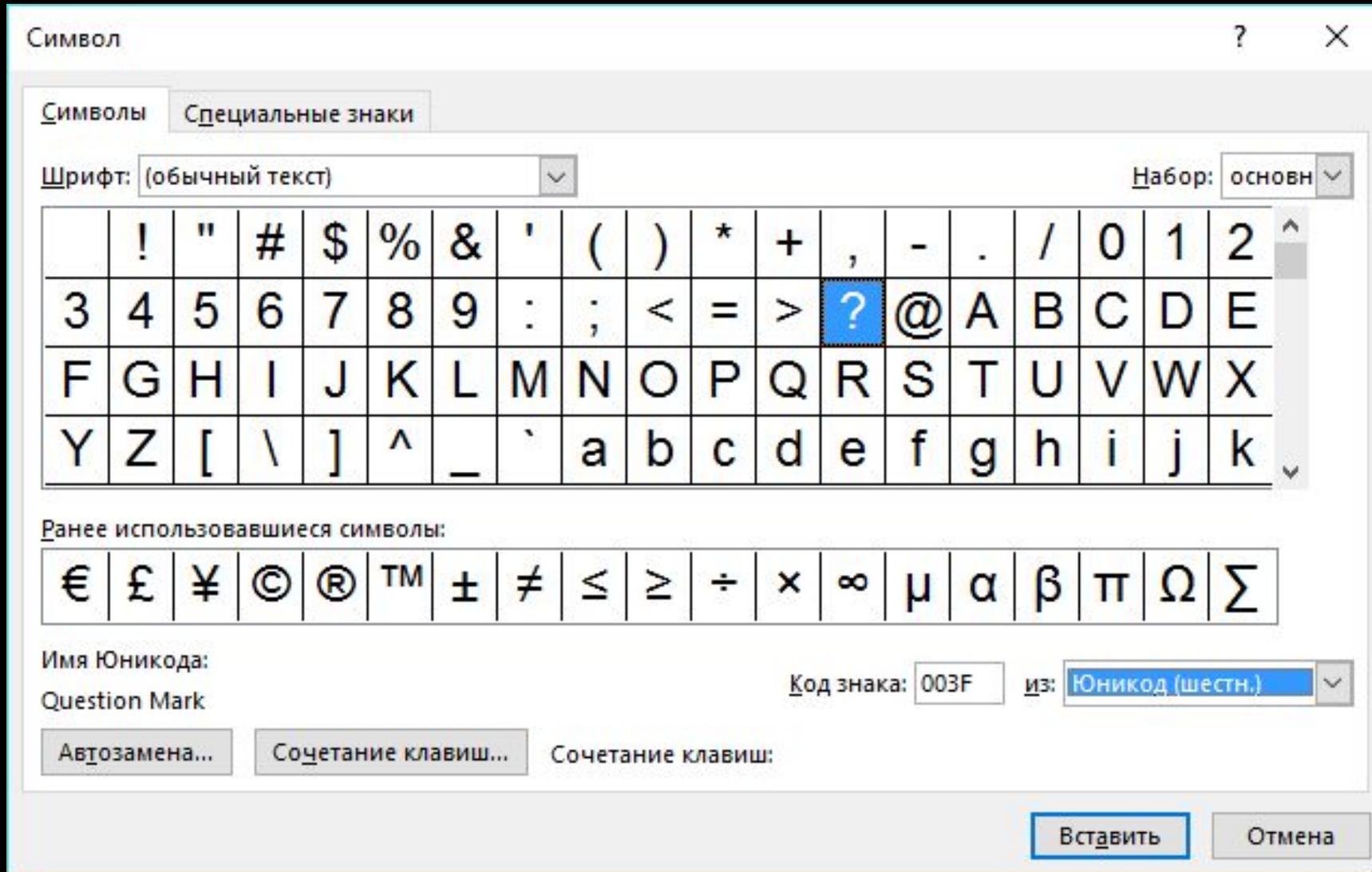
Заяц

(\ _____ /)
 (= ' . ' =)
 (") _ (")



Бабочка. Флора Стэйси. 1898 год.

Unicode – UTF-16



Кодировка текста

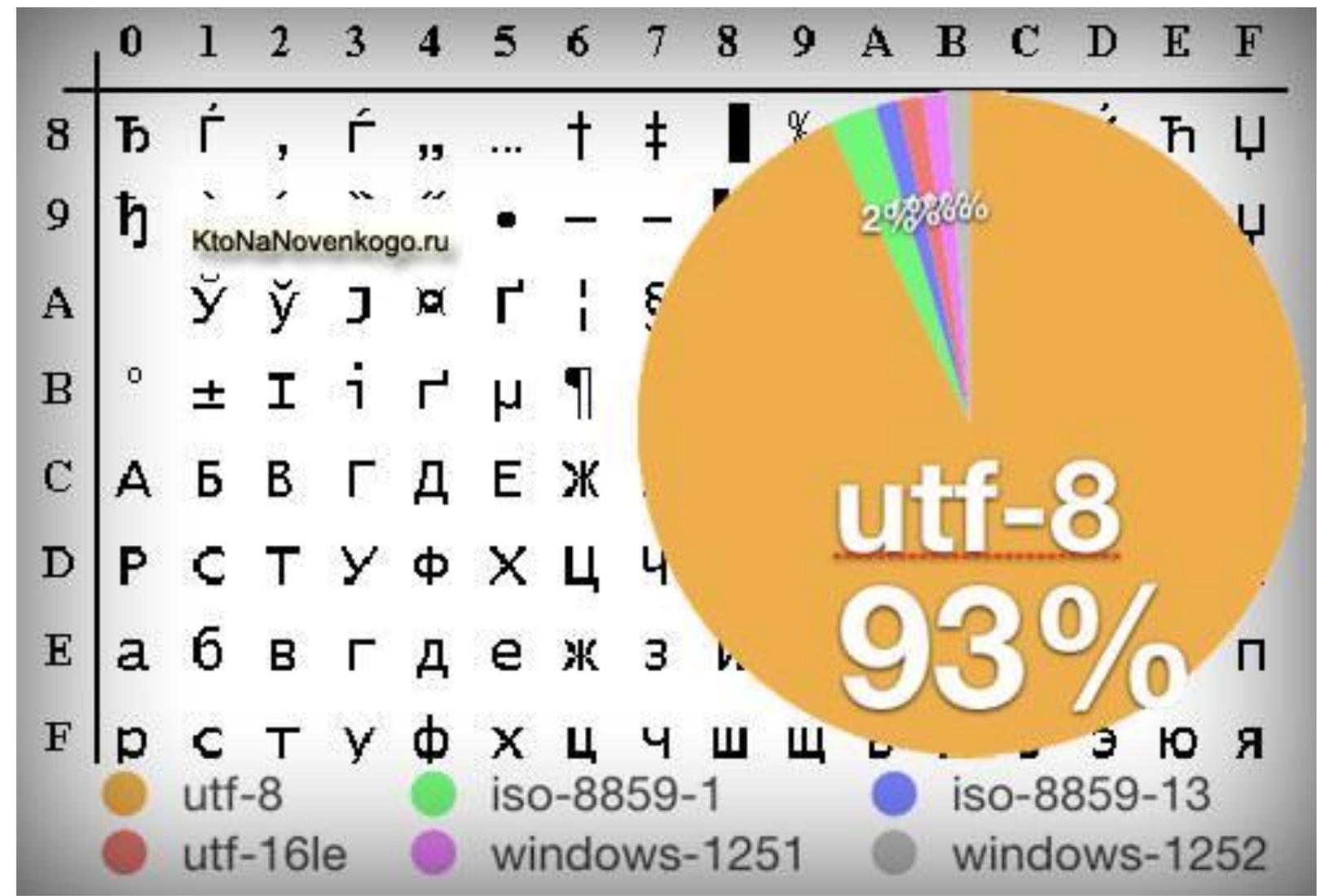
ASCII

(Windows 1251, CP866, KOI8-R)

и

Юникод

(UTF 8, 16, 32)



Формат ФЗ (с фиксированной запятой)

Число с фиксированной запятой — формат представления вещественного числа в памяти ЭВМ в виде целого числа.

Fixed Point

В англоязычном варианте знаком, отделяющим целую часть числа от дробной, является точка. Поэтому соответствующий формат называется «*Fixed Point*» или *FP*.

Число x и его целочисленное представление x' связаны формулой:

$$x = x' z$$

где z — цена (вес) младшего разряда.

Числа с фиксированной точкой (

Экспоненциальная форма записи числа, т.н. «научная нотация»
(scientific notation)

$$\begin{aligned}0.125 \cdot 10^{-2} &= 0.00125 \\0.125 \cdot 10^1 &= 1.25 \\0.125 \cdot 10^3 &= 125. \\0.125 \cdot 10^5 &= 12500.\end{aligned}$$

Информативными (несущими информацию) являются две составляющие: **мантисса** и **порядок**.

Числа с фиксированной точкой (

$$\begin{array}{r}
 +1010 \\
 0011 \\
 \hline
 1101
 \end{array}$$

$$\begin{array}{r}
 +101,0 \\
 001,1 \\
 \hline
 110,1
 \end{array}$$

$$\begin{array}{r}
 +10,10 \\
 00,11 \\
 \hline
 11,01
 \end{array}$$

$$\begin{array}{r}
 +1,010 \\
 0,011 \\
 \hline
 1,101
 \end{array}$$

$$\begin{array}{r}
 +,1010 \\
 ,0011 \\
 \hline
 ,1101
 \end{array}$$

$$\begin{array}{r}
 +1010000 \\
 0011000 \\
 \hline
 1101000
 \end{array}$$

$$\begin{array}{r}
 +0,001010 \\
 0,000011 \\
 \hline
 0,001101
 \end{array}$$

Запятая не нужна (внутри)

Форматы ФЗ/ПЗ

$$n * 2^e$$

Если экспонента обозначающая порядок (степень) -переменная, записываемая в регистр и неизвестная при компиляции, то число - с ПЗ

Если экспонента определена, и является константой, то число - с ФП

Числа с ФП могут записываться путем сохранения только мантиссы.

Зачастую в литературе экспонента - q.

“q15 multiplier” - число с ФП и экспонентой, равной 15.

Формат сложения с ФЗ

$$a = n1 * 2^{-q1} \quad b = n2 * 2^{-q2}.$$

$$a + b = n1 * 2^{-q1} + n2 * 2^{-q2} = (n1 + n2 * 2^{(q1 - q2)}) * 2^{-q1}$$

$2^{(q1 - q2)}$ - арифметический сдвиг для приведения чисел к единой экспоненте.

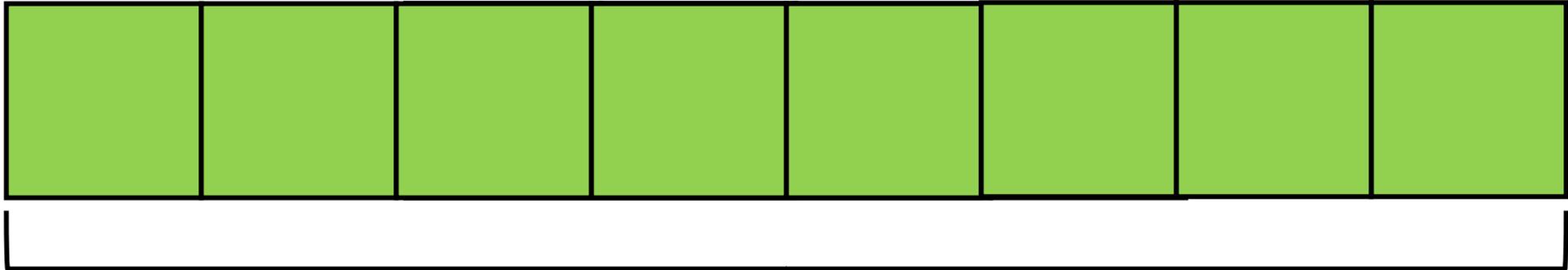
Формат умножения с ФЗ

$$a = n1 * 2^{-q1} \quad b = n2 * 2^{-q2}$$

$$a * b = n1 * 2^{-q1} * n2 * 2^{-q2} = n1 * n2 * 2^{-(q2 + q1)}$$

$2^{-(q2 + q1)}$ - экспоненты чисел при умножении складываются

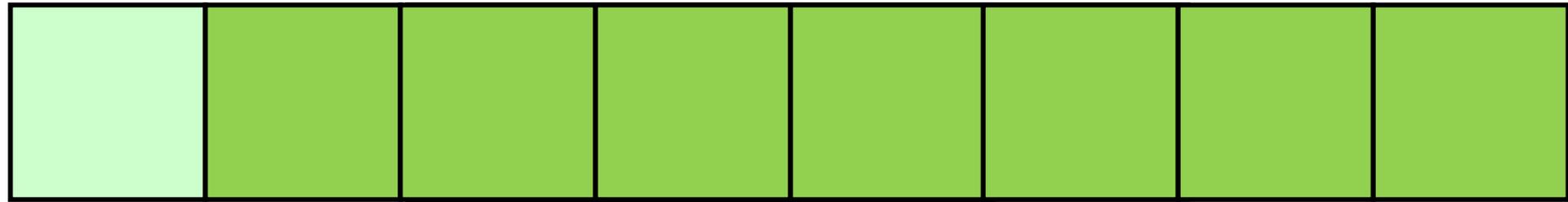
Формат числа без знака



Цифры

Формат 8 разрядов (8)

Формат числа со знаком



Знак

Цифры

Формат 8 разрядов (1+7)

Числа с фиксированной точкой (

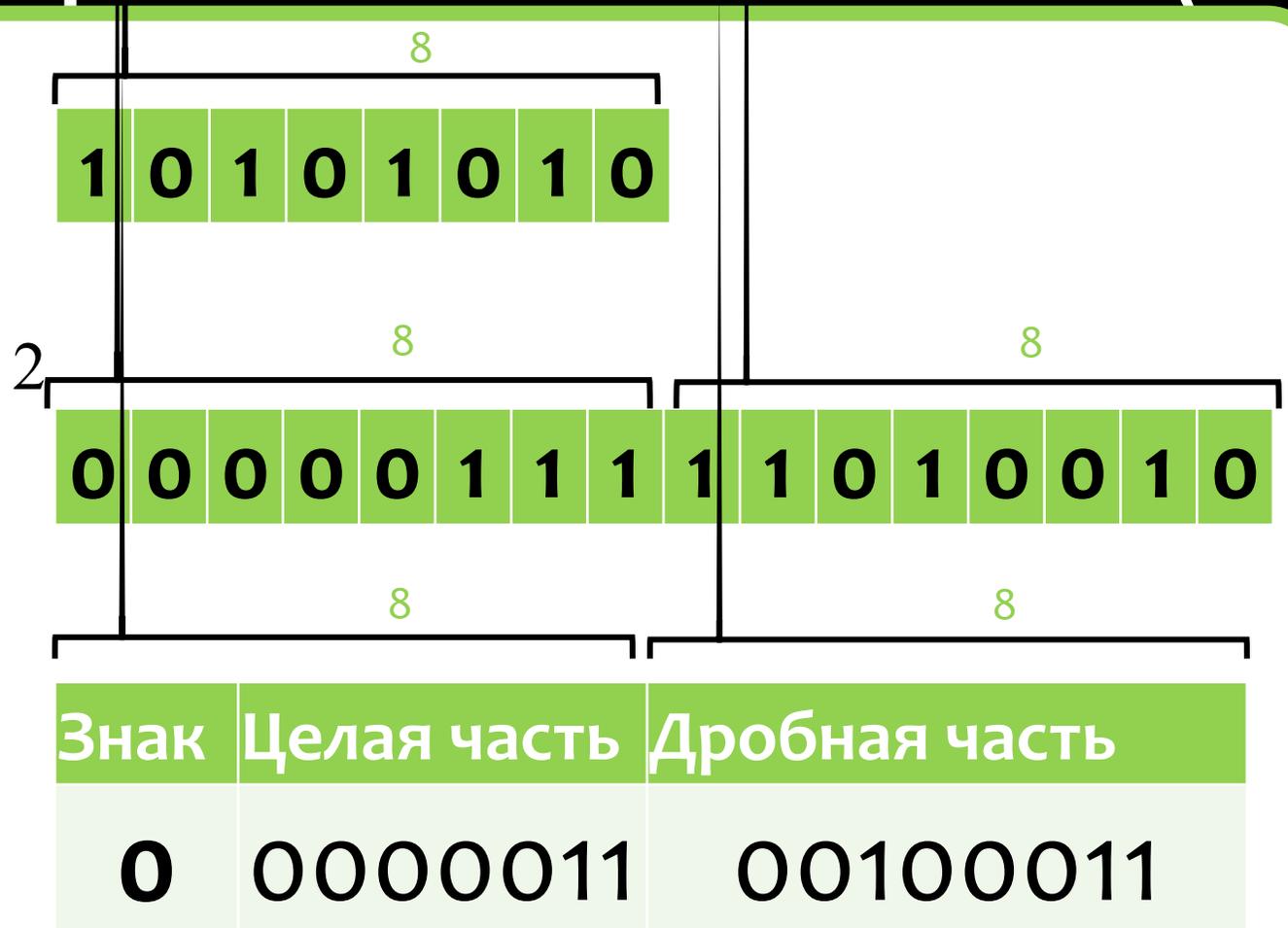
$$A_2 = 10101010_2$$

$$2002_{10} = 11111010010_2$$

$$\pi = 3,14159265359$$

$$3 = 11_2$$

$$0,14 = 0,00100011_2$$



Переполнение с ФЗ

Для 3-ех разрядов: $0,12 \times 0,12 = 0,0144$

В ФЗ 3-ех разрядов:

$$(1,20 \cdot 10^{-1}) \times (1,20 \cdot 10^{-1}) = (1,44 \cdot 10^{-2})$$

$$0,120 \times 0,120 = 0,014$$

Мы потеряли крайний правый разряд числа, так как формат ФЗ не позволяет запятой «плавать» по записи числа

Формат ПЗ (с плавающей запятой)

Floating Point

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

Так как в некоторых странах при записи чисел целая часть отделяется от дробной точкой, а в других запятой, для обозначения данного понятия используется термин «плавающая запятая» или «плавающая точка».

В настоящее время можно встретить оба варианта.

Используемое наиболее часто представление утверждено в стандарте:

The logo for the IEEE 754 standard, consisting of the text "IEEE 754" in a bold, sans-serif font, enclosed in a rectangular box with a 3D effect and a gradient background.

IEEE 754

Числа с плавающей запятой

Формат ПЗ предусматривает деление разрядной сетки на разряды мантиссы и разряды порядка (со знаками).

Если число содержит в мантиссе первыми значащими цифрами нули, то для сохранения точности нужно иметь больше разрядов. При сохранении разрядности происходит *потеря точности мантиссы*.

$$0.127 * 10^3 = 127$$

$$0.0127 * 10^4 = 127 \approx 130 \text{ (или } \approx 120)$$

$$0.013 * 10^4 = 130$$

Числа с плавающей запятой

Числа, мантисса которых содержит впереди незначащие нули, называются денормализованными.

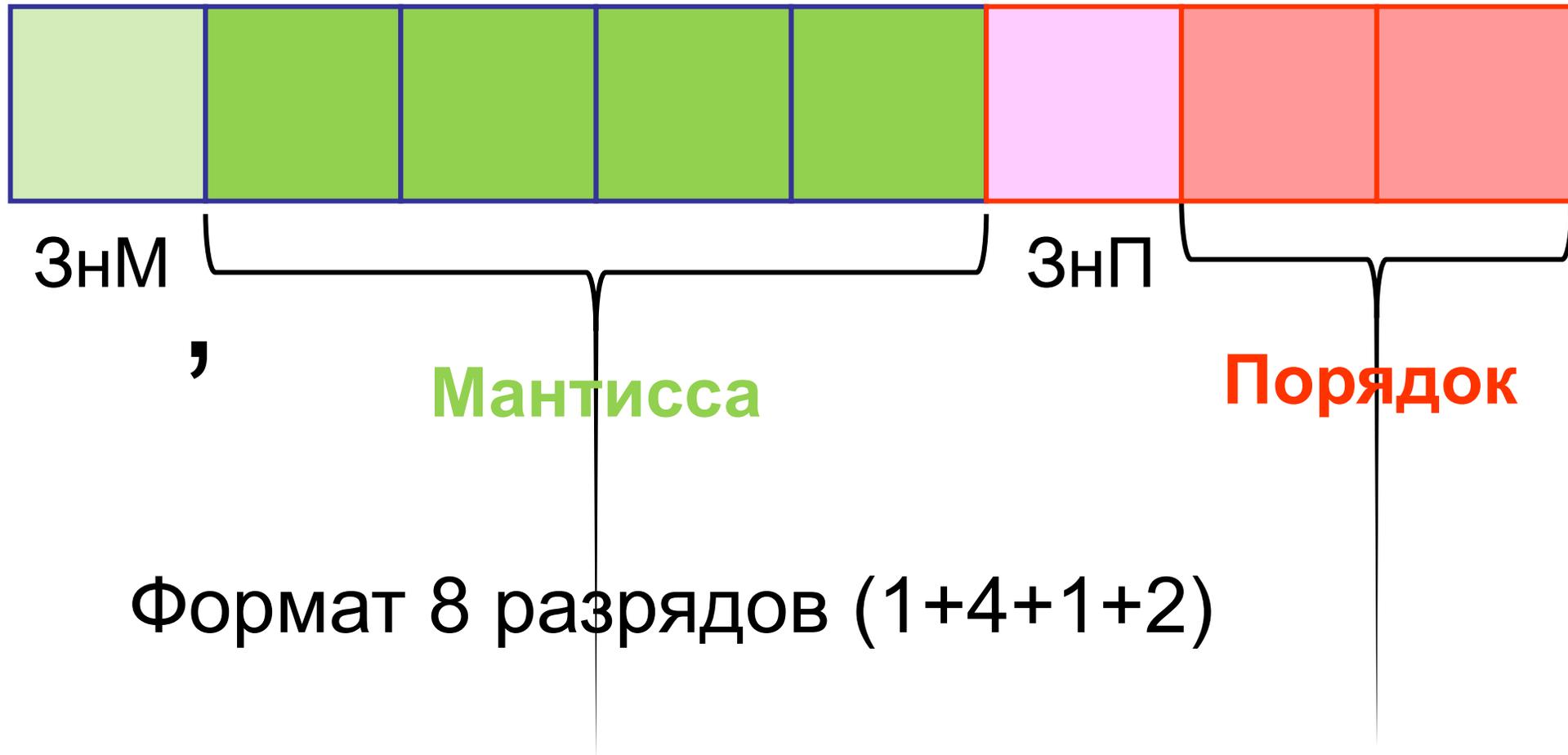
Соответственно, числа, мантисса которых не содержит впереди незначащих цифр, называются нормализованными.

Числа в памяти компьютера хранятся в нормализованном виде.

В двоичной системе счисления первая значащая цифра для положительных чисел – это единица. Отсюда следует, что в нормализованной мантиссе первая цифра всегда равна единице.

В IEEE 754 в память эта единица не записывается, и поэтому она называется «скрытой единицей».

Формат числа ПЗ (пример)



Формат числа ПЗ (пример)



+

,

0,625

+

2

$$+0,625 * 2^{+2} = +0,625 * 4 = +2,5$$

ИЛИ

$$+0,1010(2^{+2}) = +10,10 = +2,5$$

Формат ПЗ IEEE 754

Используемое наиболее часто представление числа с плавающей запятой утверждено в стандарте:



IEEE 754

Стандарт IEEE 754

IEEE 754 (с 1985) предусматривает два основных формата для представления чисел с плавающей запятой:

32-разрядные и 64-разрядные

Формат отводит самый старший бит под знак мантиссы, затем следует смещенный порядок (характеристика), потом мантисса (без скрытой единицы).

Для 32-битного формата это 1+8+23 разряда (смещение порядка 127)

Для 64-битного 1+11+52 (смещение 1023)

Стандарт IEEE 754

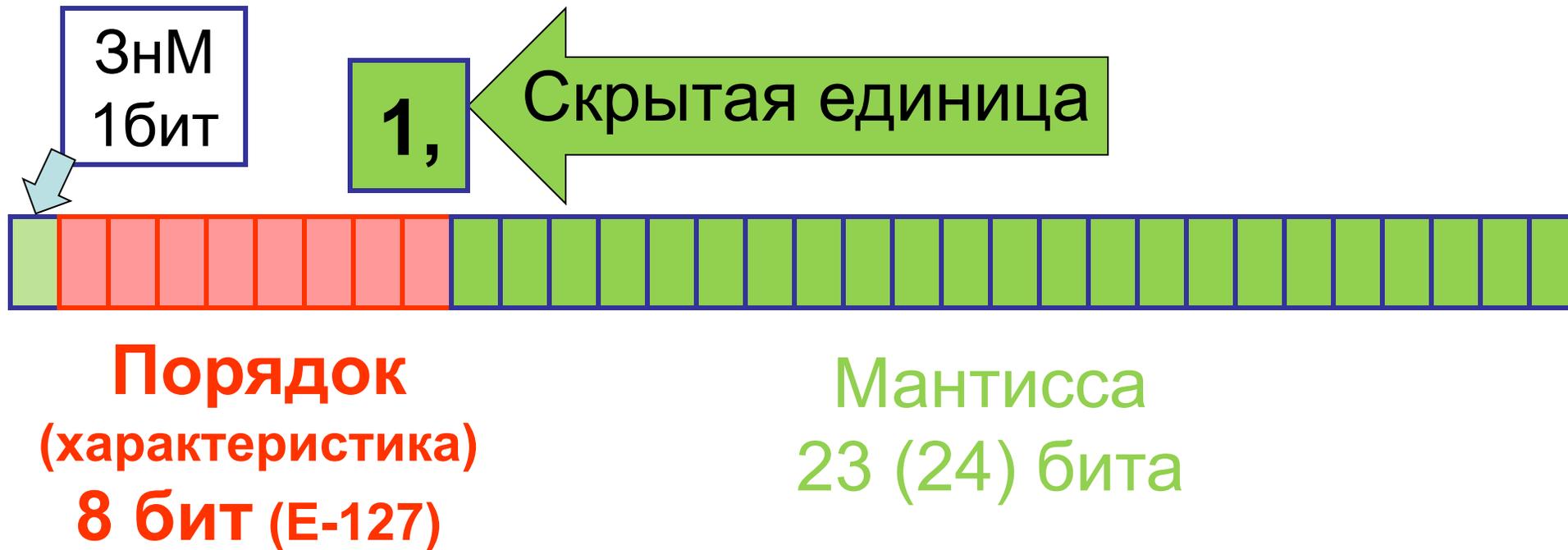
- **определяет:** как представлять нормализованные положительные и отрицательные числа с ПЗ
- как представлять денормализованные положительные и отрицательные числа с ПЗ
- как представлять нулевые числа
- как представлять специальную величину бесконечность (Infinity)
- как представлять специальную величину "Не число" (NaN или NaNs)
- четыре режима округления

IEEE 754 форматы чисел

ПЗ IEEE 754 определяет четыре формата представления чисел с плавающей запятой:

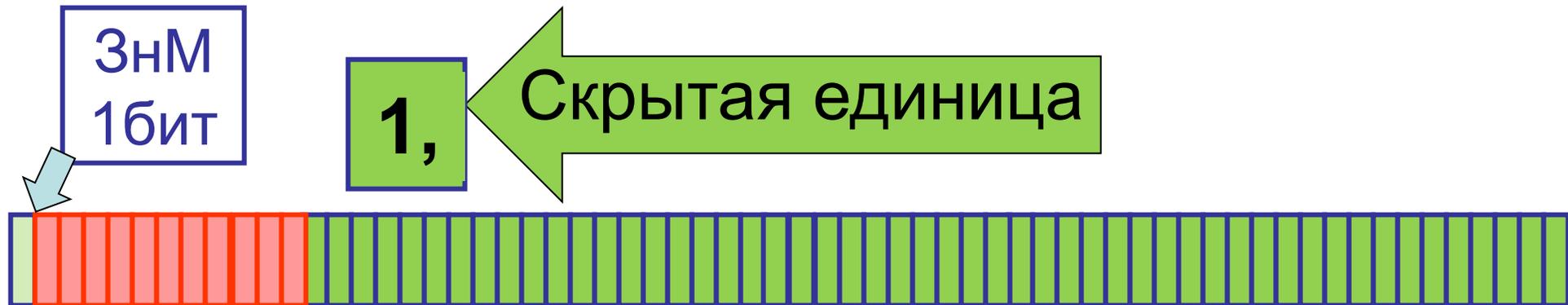
- с одинарной точностью (single-precision) 32 бита
- с двойной точностью (double-precision) 64 бита
- с одинарной расширенной точностью (single-extended precision) ≥ 43 бит (редко используемый)
- с двойной расширенной точностью (double-extended precision) ≥ 79 бит (обычно используют 80 бит)

Формат числа IEEE 754



Формат 32 разряда (1+8+23(24))

Формат числа IEEE 754



Порядок
(характеристика)
11 бит

Мантисса
52 (53) бита

Формат 64 разряда (1+11+52(53))



- это 32 бита для сравнения

Пример числа IEEE 754

Формат числа одинарной точности (float/single) 32 разряда
(1+8+23(24))



Знак $s=0$ (положительное число)

Порядок $e=0111100_2 - 127_{10} = -3$

Мантисса $M = 1.01_2$ (первая единица не явная)

Число $F = 1.01_2 e^{-3} = 2^{-3} + 2^{-5} = 0,125 + 0,03125 = 0,15625$

или

Число «101» со сдвигом запятой на несколько разрядов влево (смещение задает e).

$1,01$ — это двоичное представление, означающее $1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$.

Сдвинув запятую на три позиции влево ($e=-3$) получим:

$1,01e^{-3} = 1 \times 2^{-3} + 0 \times 2^{-4} + 1 \times 2^{-5} = 1 \times 0,125 + 0 \times 0,0625 + 1 \times 0,03125 = 0,125 + 0,03125 = 0,15625$.

Стандарт IEEE 754

Диапазон чисел с плавающей запятой:

32-разрядные – $\approx \pm 10^{\pm 38}$

64-разрядные – $\approx \pm 10^{\pm 308}$

Формат 32 бита называется
«одинарная точность» (*single precision*).

Формат 64 бита называется
«двойная точность» (*double precision*).

Позже были введены 16 бит и 128 бит (половинная и учетверенная точность).

Стандарт IEEE 754-2008

В 2008 году ассоциация IEEE выпустила стандарт IEEE 754-2008 (этому стандарту идентичен по содержанию ISO/IEC/IEEE 60559:2011), который включил в себя стандарт IEEE 754-1985.

В новом стандарте IEEE754-2008 добавлены числа:

половинной (half precision) (16 бит),

четверной (quadruple precision) (128 бит), и

расширенной точности (extended precision) (80 бит).

Также кроме чисел с основанием 2 присутствуют числа с основанием 10, так называемые десятичные (decimal) числа с плавающей запятой

Формат ФЗ

- используется ПК или ДК (чаще),
- числа расположены равномерно по числовой оси
- дискретность (расстояние между соседними числами) постоянная на всем диапазоне.

Формат ПЗ

- используется специальный формат представления чисел
- числа расположены неравномерно по числовой оси
- дискретность (расстояние между соседними числами) переменная на всем диапазоне. Чем больше значения чисел, тем больше промежутки между ними.

Типы данных IEEE 754

Четыре стандартных формата									
Формат	"Точность" precision	всего бит	e	m	d	ϵ	min денорм.	min норм.	max норм.
binary16	половинная half	16	5	10	5	$9.77 \cdot 10^{-4}$	$5.96 \cdot 10^{-8}$	$6.1 \cdot 10^{-5}$	65504
binary32	одинарная single	32	8	23	9	$1.19 \cdot 10^{-7}$	$1.4 \cdot 10^{-45}$	$1.18 \cdot 10^{-38}$	$3.4 \cdot 10^{38}$
binary64	двойная double	64	11	52	17	$2.22 \cdot 10^{-16}$	$4.94 \cdot 10^{-324}$	$2.2 \cdot 10^{-308}$	$1.8 \cdot 10^{308}$
binary128	четверная quadruple	128	15	112	36	$1.93 \cdot 10^{-34}$	$6.48 \cdot 10^{-4966}$	$3.4 \cdot 10^{-4932}$	$1.2 \cdot 10^{4932}$

Типы данных C++

Тип	байт	Диапазон принимаемых значений
целочисленные типы данных (ФЗ)		
short int	2	-32 768 / 32 767
unsigned short int	2	0 / 65 535
int	4	-2 147 483 648 / 2 147 483 647
unsigned int	4	0 / 4 294 967 295
long int	4	-2 147 483 648 / 2 147 483 647
unsigned long int	4	0 / 4 294 967 295
типы данных с плавающей точкой (ПЗ)		
float	4	-2 147 483 648.0 / 2 147 483 647.0
long float	8	-9 223 372 036 854 775 808 .0 / 9 223 372 036 854 775 807.0
double	8	-9 223 372 036 854 775 808 .0 / 9 223 372 036 854 775 807.0

Формат ПЗ: Пример кода

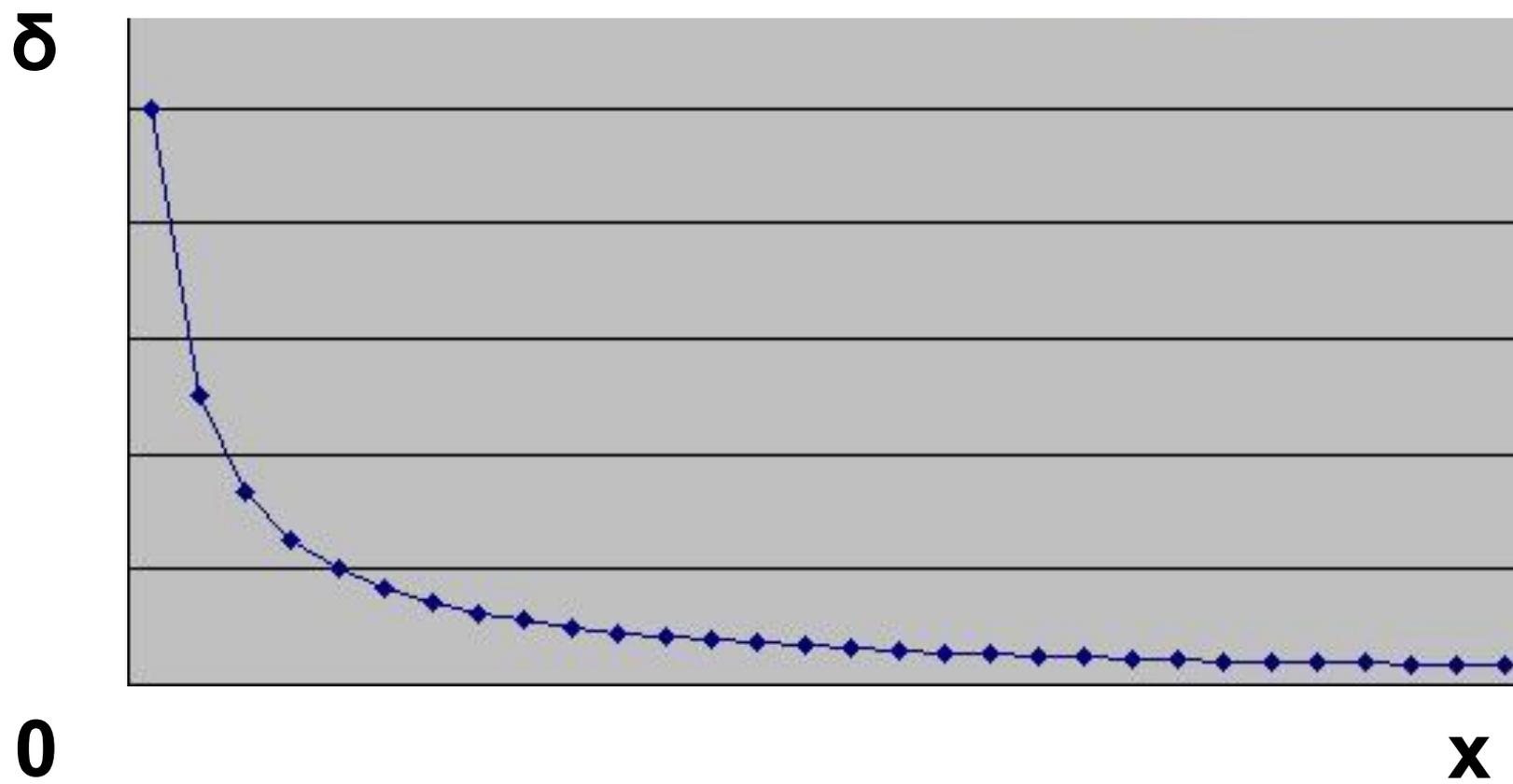
```
double a = 2.0 - 1.1;
```

```
double f = 0.0;  
for (int i=1; i <= 10; i++)  
{  
    f += 0.1;  
}
```

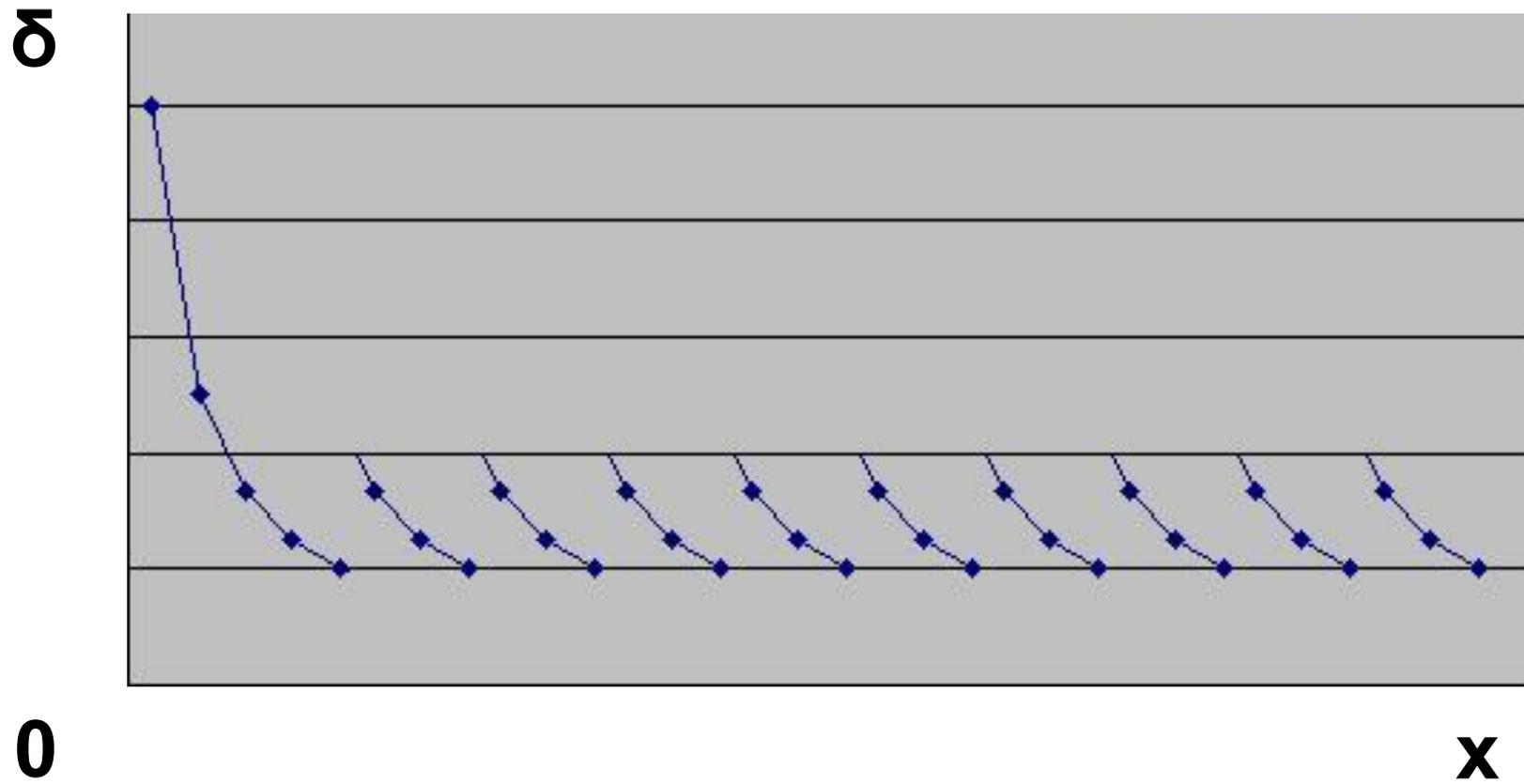
```
<< ... ?
```

```
<< ... ?
```


Относительная погрешность Φ_3



Относительная погрешность ПЗ



ТСИС

(Технические средства информационных систем)

Программное обеспечение информационных систем (1-40 01 73)



- Лекция 2

Формат с плавающей запятой.

Стандарт IEEE 754.

Погрешности.

Обратная польская запись

Ковалевский Вячеслав Викторович

4096tb@gmail.com

Тема письма:

БГУИР.