Курс: Элементы компьютерной математики

Лектор – Склярова Елена Александровна

Лекция №7

Тема: Элементы компьютерной математики (ЭКМ)

III. Элементы машинной арифметики

1.Коды для представления чисел со знаком
2.Формы представления чисел
3.Диапазон и точность представления чисел
4.Сложение и вычитание чисел с фиксированной запятой
5.Умножение и деление чисел с фиксированной запятой
6.Десятичные операции

Коды для представления чисел со знаком

Чисел без знака (ЧБЗ), конечно, недостаточно для обеспечения вычислительных работ. Естественное же представление знаков «+» и «-» годится только для ввода-вывода.

Например, можно записать:

$- 45 = - 55_8 = - 101101_2$ и т.п.

При вычислениях знак числа кодируют. Обычно так: код знака «плюс» - это 0, знак «минус» - 1.

Коды для представления чисел со знаком

Для представления чисел со знаком принято использовать три таких **специальных кода**:

- прямой код;
- обратный код;
- дополнительный код.

Коды для представления чисел со знаком

Проще всего записываются числа в прямом коде:

$(45)_{пр} = 0.45 = 0.55_8 = 0.101101_2$,

$(– 45)_{пр} = 1.45 = 1,55_8 = 1.101101_2$,

$(– 45,5)_{пр} = 1.45,5 = 1.55,4_8 = 1.101101,1_2$.

Точка «.» в записи прямою кода отделяет знаковый разряд от цифровых разрядов.

Коды для представления чисел со знаком

Правило получения прямого кода: цифровые разряды числа не изменяются, знаковый разряд отделяемся от них точкой.

!!!, для **положительных** чисел все три кода **совпадают**. Поэтому формируем правило получения обратного кода для **отрицательных** чисел: цифровые разряды двоичного числа инвертируются.

Коды для представления чисел со знаком

Примеры.

$(-45)_{обр} = (-101101_2)_{обр} = 1.010010_2$,

$(-45{,}5)_{обр} = (-101101{,}1_2)_{обр} = 1.010010{,}0_2$.

Если система не двоичная ($q \neq 2$), действует общее правило: каждая цифра дополняется до значения $(q - 1)$.

Коды для представления чисел со знаком

Примеры.

$(-45)_{обр} = 1.54 = 1.22_8$

$(-45{,}5)_{обр} = 1.54{,}4_{обр} = 1.22.3_8$

Нуль в прямом и обратном кодах имеет двоякое представление:

$(+0)_{пр} = 0.0 \ldots 0,$

$(-0)_{пр} = 1.0 \ldots 0,$

$(+0)_{обр} = 0.0 \ldots 0,$

$(-0)_{обр} = 1.1 \ldots 1_2 = 1.9 \ldots 9 = \ldots$

Дополнительный код отрицательного числа может быть получен прямо или косвенно (через обратный код).

Коды для представления чисел со знаком

Прямое правило:

цифровые разряды отрицательного числа инвертируются, за исключением самой правой единицы и, возможно, стоящих за ней нулей (эта единица и нули не изменяются).

Коды для представления чисел со знаком

Примеры.

$(-\ 45)_{доп} = (-\ 101101_2)_{доп} = 1.010011_2$,

$(-\ 45{,}5)_{доп} = (-\ 101101{,}1_2)_{доп} = 1.010010{,}1_2$,

$(-\ 10)_{доп} = (-\ 1010)_{2\ доп} = 1.0110_2$.

Общее правило для системы с основанием q:

каждая цифра дополняется до значения (q - 1), за исключением самой правой значащей цифры и, возможно, стоящих за ней нулей (эта цифра дополняется до значения q, а нули не изменяются).

Коды для представления чисел со знаком

Примеры.

$(-45)_{доп} = 1.55 = 1.23_8$

$(-45{,}5)_{доп} = 1.54{,}5 = 1.22.4_8$

$(-10)_{доп} = 1.90$

Коды для представления чисел со знаком

Косвенное правило: к обратному коду отрицательного числа надо добавить единицу в младшем разряде.

Интересной особенностью дополнительного кода является наличие единственного кода нуля:

$(0)_{доп} = (+\ 0)_{доп} = 0.0 \ldots 0,$

Это следует из косвенного правила для (– 0):

$(-\ 0)_{доп} = (-\ 0)_{обр} + 1 = 1.1 \ldots 1_2 + 1 = [1]\ 0.0 \ldots 0_2.$

Здесь в сложении участвуют все разряды, включая знаковый.

Коды для представления чисел со знаком

Невостребованность кодовой комбинации для (– 0) позволяет несколько расширить диапазон значений, представимых в дополнительном коде.

Наибольшее по абсолютной величине отрицательное число имеет при общем количестве цифровых разрядов дополнительного кода n значение (-2^n):

$$(-2^n)_{доп} = (-1\ 0 \ldots 0_2) = 1.0 \ldots 0_2.$$

□□□ □□□

n n

Коды для представления чисел со знаком

Это следует хотя бы из логики такой числовой последовательности:

$(-\ 6)_{доп} = (-\ 110_2)_{доп} = 1.010_2$

$(-7)_{доп} = (-111_2)_{доп} = 1.001_2$

$(-8 = -2^3)_{доп} = (-10002)_{доп} = 1.000_2$

Здесь справа – последовательные убывающие двоичные числа (точка-разделитель игнорируется).

Каждый из трех видов кода имеет модификацию.

В модифицированном коде – не один, а два знаковых разряда. Они имеют одинаковые значения (00 или 11).

Формы представления чисел в ЭВМ

Классификацию числовых форматов можно провести по трем признакам:

– основание системы счисления;

– наличие дробной части (целые или дробные числа);

– наличие экспоненциального множителя (числа с фиксированной или плавающей запятой).

Формы представления чисел в ЭВМ

В ЭВМ используются обычно 3 – 4 формата:

- целые числа (двоичные; запятая фиксирована после младшего разряда);
- числа с фиксированной запятой (двоичные; дробные; запятая фиксирована после знакового разряда);
- числа с плавающей запятой (двоичные; дробные; имеются мантисса и порядок – показатель степени основания системы счисления);
- десятичные числа (целые; запятая фиксирована после младшего разряда).

Формы представления чисел в ЭВМ

В современных ЭВМ «классический формат» с фиксированной запятой не используется. Его роль вполне реализует формат целых чисел (рис.1).

Кстати, при выполнении арифметических операций разница между этими форматами проявляется только на уровне умножения и деления. Код – дополнительный.

±	Цифровые разряды

15 14 0

Рис. 1. Пример формата «целые числа»

Формы представления чисел в ЭВМ

Двоичные числа с плавающей запятой (рис. 2) имеют мантиссу (m_x) и порядок (p_x):

$$X = m_x * 2^{Px}$$

±	Характеристика	Мантисса (цифровые разряды)
31	30 … 24	23 … 0

Рис. 2. Пример формата с плавающей запятой

Формы представления чисел в ЭВМ

Мантисса числа – это правильная дробь ($|m_x| < 1$), представлена в прямом коде Знаковый разряд ее, или, что то же, знаковый разряд числа, – разряд {31}. Количество цифровых разрядов мантиссы в примере – 24.

Характеристика представляет собою число без знака (≥ 0), а именно – порядок, смешенный в неотрицательную область:

$$H_x = p_x + 64 = 0...127,$$

$$p_x = H_x - 64 = -64 \ldots 63.$$

Формы представления чисел в ЭВМ

Выполнение действий +/– над порядками, представленными в дополнительном коде, практически равнозначно аналогичным действиям над характеристиками. Способ кодирования знака при этом особой роли не играет. Сложение знаковых разрядов, правда, нужно «инвертировать» (вместо ⊕ реализуется ≡).

№	Порядок p (доп. код)		Характеристика H = p + 8	
1	7	0.111	1111	15
2	6	0.110	1110	14
…	…	…	…	…
7	1	0.001	1001	9
8	0	0.000	1000	8
9	–1	1.111	0111	7
…	...	…	…	…
15	–7	1.001	0001	1
16	–8	1. 000	0000	0

Формы представления чисел в ЭВМ

Наибольшей точности числа с плавающей запятой соответствует его **нормализованное** представление:

$$2^{-1} \le |m_x| < 1.$$

Таким образом, старшая двоичная цифра мантиссы должна быть единицей.

Формы представления чисел в ЭВМ

Десятичные числа в старых «больших» машинах (ЕС ЭВМ) представлены полями переменной длины – от 1 до 16 байтов. Ввод-вывод их осуществляется в распакованном (неупакованном, зонном) Z-формате (рис3.а), а обработка – в упакованном P-формате (рис. 3б).

Зона	Цифра	Зона	Цифра	…	Знак	Цифра

0 3 4 7 …

Цифра	Цифра	Цифра	Цифра	…	Цифра	Знак

0 3 4 7 …

Рис. 3. Форматы десятичных чисел

Формы представления чисел в ЭВМ

«Зона» в неупакованном формате – это 1111_2 = F_{16}.

Вместе с последующей двоичной тетрадой, представляющей десятичную цифру, зона образует байт символа, кодируемого в ДКОИ («Двоичный код обмена информацией»).

Код знака (в последнем, младшем байте) C, E или F_{16} для « + » и D_{16} для « – ».

В упакованном формате каждый байт, кроме последнего, содержит 2 десятичных цифры. Это означает, что десятичный операнд может иметь от 1 до 31 разряда.

Формы представления чисел в ЭВМ

Код для чисел со знаком – **прямой**.

Самое правое положение тетрады знака благоприятствует побайтному (последовательно-параллельному) выполнению арифметической операции, начинающейся с **младших** разрядов операндов.

В алгебраическом сложении используется **дополнительный** код, и для преобразования отрицательных операндов и результатов «прямой-дополнительный-прямой» требуется значительное время.

Диапазон и точность представления чисел

Диапазон представления целых ч и с е л, заданных в формате {0:n} (n – количество цифровых разрядов, равное 15 для случая рис. 1), определяется двояко:

$$X_{min} \le X \le X_{max}$$

$$0 \ne |X|_{min} \le |X| \le |X|_{max}$$

Учитывая особенность представления максимальных по абсолютной величине отрицательных чисел в дополнительном коде, получаем:

$$X_{min} = -2^n, \; X_{max} = -2^n-1$$

$$|X|_{min} = 1, \; |X|_{max} = 2^n,$$

Диапазон и точность представления чисел

Для n = 15 (рис.1) находим:

$$-2^{15} = -32\ 768 \le X \le 2^{15} - 1 = 32\ 767,$$

$$1 \le |X| \le 32\ 768.$$

Машинное представление здесь таково:

$(X_{min})_{доп} = 1.0 \ldots 0_2$

□□□

n

$(X_{max})_{доп} = 0.1 \ldots 1_2$

□□□

n

Диапазон и точность представления чисел

Точность представления чисел связывается обычно с количеством значащих цифр (двоичных, десятичных, ...).

Для **целых** форматов оценка этой точности фактически **равнозначна** оценке диапазона. Она определяется n двоичными разрядами.

Для получения более привычной десятичной оценки можно воспользоваться естественным соотношением:

$$2^x \approx 10^y,$$

$$X \lg 2 \approx y,$$

$$y \approx 0{,}3010\ x \approx 0{,}3\ x.$$

Десятичная точность целых форматов – **0,3n.**

Например, 15 x 0,3 = 4,5.

Диапазон и точность представления чисел

Диапазон для чисел с плавающей запятой абсолютно симметричен (в силу прямого кода мантиссы):

$$|X_{min}| = |X_{max}| = |X|_{max},$$

Поэтому здесь интерес представляет только диапазон для модуля:

$$|X|_{min\ норм} \leq |X| \leq |X|_{max}.$$

Индекс «норм» означает **нормализованность** чисел с плавающей запятой:

$$2^{-1} \leq |m_x| < 1.$$

Старшая двоичная цифра мантиссы должна быть 1.

Диапазон и точность представления чисел

$|X|_{\text{min норм}} \leq 2^{-1} * 2^{-64} = 2^{-65} \approx 10^{-19}.$

$|X|_{\max} = (1 - 2^{-n_m}) * 2^{63} \approx 2^{63} \approx 10^{19}.$

Здесь n_m – количество двоичных цифровых разрядов мантиссы

(на рис.2 их 24).

!!! Разрядность мантиссы существенно определяет точность чисел с плавающей запятой.

Диапазон и точность представления чисел

Значащие цифры числа, независимо от его представления, – это значащие цифры **мантиссы**.

24-разрядная мантисса (рис. 2) соответствует точности 7 десятичных цифр.

Диапазон и точность представления десятичных чисел, как и чисел с фиксированной запятой (в частности, целых), оцениваются одинаково – длиной формата. Оценка для симметричного диапазона в случае упакованного 16-байтного формата (рис. 3):

$$0 \neq 1 \leq |x| \leq 10^{31} - 1,$$

для точности – 31 десятичная цифра.

Сложение и вычитание чисел с фиксированной запятой

Сложение и вычитание представляют пару операций «типа сложения», т.е. алгебраическое сложение, которое, в свою очередь, можно понимать как сложение чисел со знаком, заданных в обратном или дополнительном коде.

Вычитание может выполняться непосредственно (с использованием, например, специальных операционных элементов – вычитателей) или косвенно, путем сведения его к сложению:

$$Z := X - Y = X + (-Y)$$

Сложение и вычитание чисел с фиксированной запятой

В последнем случае достаточно, как видно, изменить знак второго операнда. Если операнды (и результат) представлены в дополнительном коде, изменение знака производится путем инверсии **всех** разрядов и добавления 1 в младшем разряде. **Например,**

$Y = \ \ 5 \sim 0.101_2$

$-Y = -5 \sim 1\,010_2$

$+ \ \underline{\quad\quad 1}$

$1.011_2 \ = (-5)_{доп},$

и наоборот,

$Y \ = -5 \sim 1.011_2$

$-Y = \ \ 5 \sim 0\,100_2$

$+ \ \underline{\quad\quad 1}$

$0.101_2 \ = (5)_{доп}.$

Сложение и вычитание чисел с фиксированной запятой

Правила алгебраического сложения чисел в обратном и дополнительном кодах тривиальны: обратные или дополнительные коды операндов суммируются как обыкновенные числа без знака, возможная единица переноса из знакового разряда (старшего знакового разряда, если код модифицированный) циклически переносится в младший разряд для второго суммирования (обратный код) или отбрасывается (дополнительный код).

« + » – знак операции сложения с циклическим переносом (Пример 1).

Сложение и вычитание чисел с фиксированной запятой

Пример 1.

$$
\begin{array}{rl}
 & X = -5 \\
+ & \\
 & Y = \ \ 7 \\
\hline
 & Z = \ \ 2
\end{array}
\qquad\qquad
\begin{array}{rr}
 & X_{обр} = 1.010_2 \\
+ & \\
 & Y_{обр} = 0.111_2 \\
\hline
 & 10.001 \\
+ & 1 \\
\hline
 & Z_{обр} = \ 0.010_2
\end{array}
$$

Сложение и вычитание чисел с фиксированной запятой

Пример 2.

$$
\begin{array}{rl}
 & X = \ \ 5 \\
+ & \\
 & Y = -7 \\
\hline
 & Z = -2
\end{array}
\qquad\qquad
\begin{array}{rl}
 & X_{\text{доп мод}} = 00.101_2 \\
+ & \\
 & Y_{\text{доп мод}} = 11.001_2 \\
\hline
 & Z_{\text{доп мод}} = 11.110_2
\end{array}
$$

Сложение и вычитание чисел с фиксированной запятой

Вообще, при сложении чисел с разными знаками в дополнительном коде отсутствие переноса свидетельствует об отрицательном результате (как в Примере 2), а наличие – о результате положительном или нулевом (Пример 3).

Пример 3.

$$\begin{array}{l} X = 5 \\ \quad + \\ \quad Y = -5 \\ \hline \quad Z = 0 \end{array} \qquad \begin{array}{l} X_{доп} = 0.101_2 \\ \; + \\ \quad Y_{доп} = 1.011_2 \\ \hline Z_{доп} = 10.010_2 \\ \qquad\qquad \llcorner p_{вых} = 1 \end{array}$$

Сложение и вычитание чисел с фиксированной запятой

В обратном коде отсутствие выходного переноса свидетельствует о неположительном результате (Пример 4), а наличие его – о результате положительном (Пример 1).

Пример 4.

$X = 5$	$X_{обр} = 0.101_2$
$+$	$+$
$Y = -5$	$Y_{доп} = 1.010_2$
$Z = 0$	$Z_{обр} = 1.111_2 = (-0)_{обр}$
	$p_{вых} = 0$

Сложение и вычитание чисел с фиксированной запятой

Обнаружение переполнения разрядной сетки при сложении может производиться несколькими способами.

Самый простой способ – использование модифицированного кода (с двумя знаковыми разрядами).

Старший знаковый разряд даже при переполнении сохраняет информацию о знаке результата («Разряд знака»).

Младший – «Разряд переполнения». Комбинация знаков при «положительном» переполнении – 01,

при «отрицательном» – 10.

Сложение и вычитание чисел с фиксированной запятой

Пример 5.

$$
\begin{array}{ll}
X = 3 & X_{доп\ мод} = 00.011_2 \\
+ & + \\
Y = 6 & Y_{доп\ мод} = 00.110_2 \\
\hline
Z = 9 > 7 & Z_{доп\ мод} \sim 01.001_2 \text{ (положительное переполнение)}
\end{array}
$$

Пример 6.

$$
\begin{array}{ll}
x = -3 & X_{доп\ мод} = 11.101_2 \\
+ & + \\
y = -6 & Y_{доп\ мод} = 11.010_2 \\
\hline
z = -9 < 8 & Z_{доп\ мод} \sim 10.111_2 \text{ (отрицательное переполнение)}
\end{array}
$$

Сложение и вычитание чисел с фиксированной запятой

В примере 6 указано граничное значение (-8), которое может быть представлено без переполнения:

$$(-8)_{\text{доп мод}} = 11.000_2$$

<u>Недостаток</u> способа модифицированного кода — расширение разрядной сетки на один разряд.

Сложение и вычитание чисел с фиксированной запятой

Второй способ обнаружения переполнения - сравнение переносов в знаковый разряд и из знакового разряда. Переполнение - при несовпадении этих переносов. Фактически здесь тоже «задействован» модифицированный дополнительный код.

Случай А. Неотрицательные операнды.

	$+$ 0	0.
	0	0.
Перенос «в»	0	X
Сумма первичная	0	0.
Сумма конечная	0	X.

Сложение и вычитание чисел с фиксированной запятой

Правило сравнения переносов дает значение признака переполнения:

$$\varphi_р = 0 \oplus X = X$$

(переполнение при X = 1).

Слева от штриховой черты показаны значения воображаемого модифицированного дополнительного кода.

Правило этого способа дает такое же значение признака переполнения:

$$\varphi_м = 0 \oplus X = X$$

Сложение и вычитание чисел с фиксированной запятой

Случай В. Отрицательные операнды.

	+ 1	1.
	1	1.
Перенос «в»	1	X
Сумма первичная	0	0.
Сумма конечная	1	X.

Здесь тоже $\varphi_р = \varphi_м = 1 \oplus X = \overline{X}$

(переполнение может быть только отрицательное - при отсутствии переноса из старшего цифрового разряда).

Сложение и вычитание чисел с фиксированной запятой

Случай С. Операнды имеют разные знаки

	$+\begin{matrix}0\\1\end{matrix}$	$\begin{matrix}0.\\1.\end{matrix}$
Перенос «в»	X	X
Сумма первичная	1	0.
Сумма конечная	$\overline{X}$	$\overline{X}$.

Оба признака переполнения снова совпадают, они имеют нулевые значения (переполнение в принципе невозможно):

$$\varphi_р = X \oplus X = 0$$

$$\varphi_м = X \oplus X = 0$$

Сложение и вычитание чисел с фиксированной запятой

Третий способ - сравнение знаков. Реализуется программно (микропрограммно). Сначала проверяется, имеют ли операнды одинаковые знаки. И, если имеют, совпадает ли с этими знаками знак результата. Переполнение соответствует несовпадению (рис. 4).

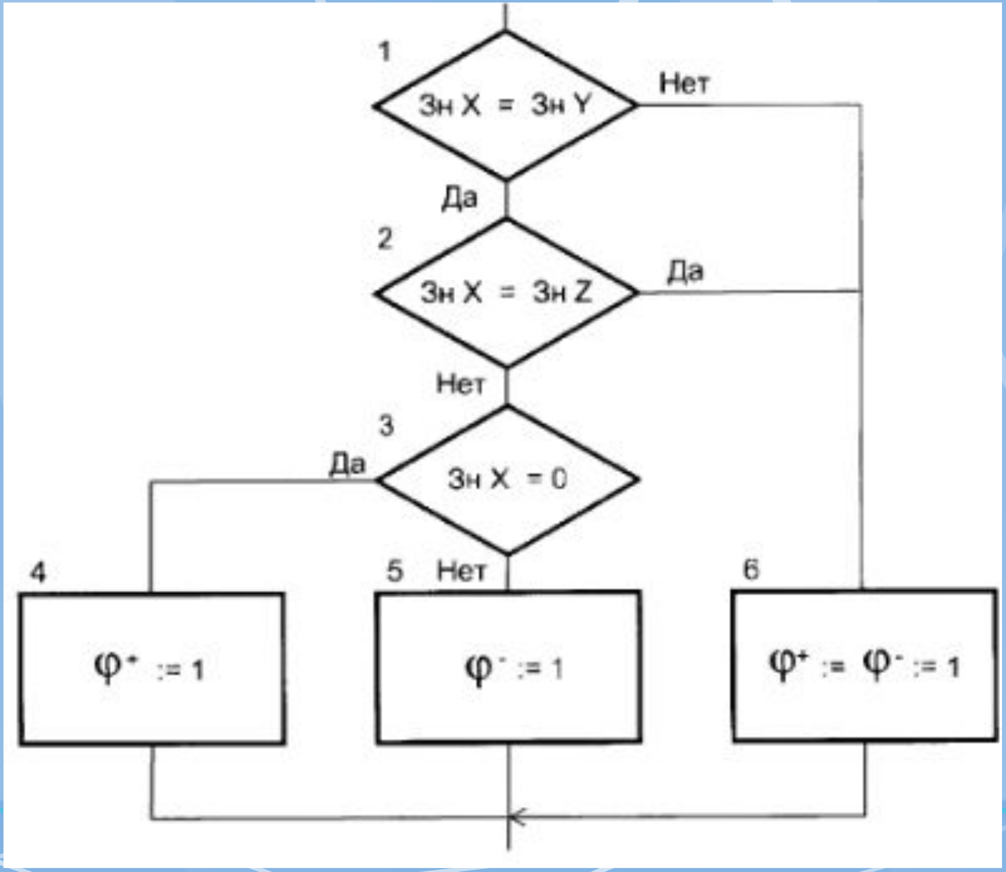

Рис. 4. Обнаружение переполнения

Лекция окончена

Нажмите клавишу <ESC> для выхода