Конспекты лекций по курсу
«Введение в информатику и системы программирования», 1 семестр
С.А. Немнюгин, направление «Прикладные математика и физика»)

Лекция 9

Архитектура ЭВМ

Форматы хранения данных. Машинная арифметика

Арифметические операции в двоичной системе счисления

Сложение

```
1 1 0 0 1 0 1 0        1 1 0 0 1 0 1 0
0 0 1 0 1 0 1 1        0 0 1 1 1 0 1 1
1 1 1 1 0 1 0 1        0 0 0 0 0 1 0 1
```

Переполнение разрядной сетки

Умножение

```
     11011
     00101
     11011
   11011  .
  10000111
```

Умножение в двоичной системе счисления = поразрядные сдвиги + суммирование

Основные форматы хранения целых чисел со знаком. Прямой код

Знак = $(-1)^s$

S	Абсолютная величина

Однобайтовый (абс. величина)
000 0000 - 111 11112
0 - 12710

Двухбайтовый (абс. величина)
0000000 00000000 – 1111111 111111112
0 – 32 76710

Четырехбайтовый (абс. величина)
0000000 00000000 00000000 00000000 –
1111111 11111111 11111111 111111112
0 – 2 147 483 64710

Обратный код

-13

Прямой код:
1000 1101$_2$

Обратный код:
1111 0010$_2$

Дополнительный код

-13

Обратный код:
1111 0010$_2$

Дополнительный код:
1111 0011$_2$

Арифметические операции в прямом, обратном и дополнительном коде

Разберем четыре случая, возникающих при вычислении суммы $x + y$. Двоичные значения в обратном коде выделены полужирным шрифтом. Единица в круглых скобках обозначает перенос разряда.

1. $x \geq 0$, $y \geq 0$. Суммируются все двоичные разряды прямого кода слагаемых, в том числе разряды знака. Например:

$$5_{10} + 9_{10} = 0000\ 0101_2 + 0000\ 1001_2 = 0000\ 1110_2 = 14_{10}.$$

2. $x \geq 0$, $y \leq 0$, $|y| > |x|$. Например:

$$5_{10} + (-9_{10}) = 0000\ 0101_2 + \mathbf{1111\ 0110_2} = \mathbf{1111\ 1011_2} = 1000\ 0100_2 = -4_{10}.$$

3. $x \geq 0$, $y \leq 0$, $|y| < |x|$. Например:

$$9_{10} + (-5_{10}) = 0000\ 1001_2 + \mathbf{1111\ 1010_2} = \mathbf{(1)0000\ 0011_2} = +3_{10}.$$

Здесь результат оказывается неправильным вследствие переноса из знакового разряда. Если единицу перенести в младший разряд, получим правильный ответ в прямом коде:

$$0000\ 0100_2 = 4_{10}.$$

4. $x \leq 0$, $y \leq 0$. Например:

$$(-9_{10}) + (-5_{10}) = \mathbf{1111\ 0110_2} + \mathbf{1111\ 1010_2} = \mathbf{(1)1111\ 0000_2} = -112_{10}.$$

Здесь результат также неправильный вследствие переноса из знакового разряда. Если единицу перенести в младший разряд, получим правильный ответ:

$$\mathbf{1111\ 0001_2} = 0000\ 1110_2 = -14_{10}.$$

Рассмотрим теперь сложение чисел в дополнительных кодах. Здесь следует рассмотреть те же четыре случая, что и для обратных кодов. Двоичные значения в обратном коде выделены полужирным шрифтом, а единица в круглых скобках обозначает перенос разряда.

1. $x \ge 0$, $y \ge 0$. Суммируются все двоичные разряды прямого кода слагаемых, в том числе старшие (разряды знака). Поскольку в данном случае используется прямой код, единственная возможная проблема — переполнение разрядной сетки.
2. $x \ge 0$, $y \le 0$, $|y| > |x|$. Например:

   $$5_{10} + (-9_{10}) = 0000\ 0101_2 + \mathbf{1111\ 0111_2} = \mathbf{1111\ 1100_2} = 1000\ 0100_2 = -4_{10}.$$

   При преобразовании дополнительного кода в прямой знаковый разряд остается неизменным, остальные разряды инвертируются (то есть единицы заменяются нулями, и наоборот) и к младшему разряду прибавляется единица.
3. $x \ge 0$, $y \le 0$, $|y| < |x|$. Например:

   $$9_{10} + (-5_{10}) = 0000\ 1001_2 + \mathbf{1111\ 1011_2} = (1)0000\ 0100_2 = 4_{10}.$$

   Здесь результат, если отбросить перенос из знакового разряда, правильный.
4. $x \le 0$, $y \le 0$. Например:

   $$(-9_{10}) + (-5_{10}) = \mathbf{1111\ 0111_2} + \mathbf{1111\ 1011_2} = \mathbf{(1)1111\ 0010_2} = 1000\ 1110_2 = -14_{10}.$$

   Результат также правильный.

Преобразование дробных значений между системами счисления

Перевод дробного числа из десятичной системы счисления в произвольную *p*-ичную:

1. умножить D_{10} на p;
2. целую часть произведения представить соответствующей цифрой *p*-ичного представления;
3. умножать D_{10} на p до тех пор, пока будет достигнута требуемая точность (получится *p*-ичное число с *k* дробными разрядами) или дробная часть обратится в ноль.

Числа с плавающей точкой

Число с фиксированной точкой:
3.14159

Число с плавающей точкой:
$(-1)^s 0.m\ p^n$
Здесь:
m – мантисса;
n – порядок.
0.314 101   0.0314 102

Машинные числа

Машинными называются числа, допускающие точное представление во внутреннем формате хранения числовых значений.

Форматы хранения чисел с плавающей точкой

Институт инженеров по электротехнике и электронике (Institute of Electrical and Electronics Engineers, IEEE) разработал международные стандарты, которые описывают представление чисел с плавающей запятой:

- стандарт ANSI/IEEE 754:1985 определяет требования к реализации двоичной плавающей арифметики;
- ANSI/IEEE 854:1987 обобщает прежний стандарт, допуская дополнительно, кроме двоичного, десятичное основание представлений мантиссы и экспоненты и произвольную длину машинного слова.

Позднее требования этих стандартов были отражены в стандарте IEC 60559:1989.
Стандарты, кроме форматов представления, описывают также основные арифметические действия, операции вычисления остатка от деления, квадратного корня, преобразования из двоичного представления в десятичное и наоборот.

В большинстве современных платформ, таких как Intel и большинстве RISC-систем аппаратно реализована плавающая арифметика, соответствующая стандарту IEC 60559.

Стандарты IEEE определяют следующие форматы хранения вещественных чисел:

- ***с простой точностью*** (соответствует типам REAL*4 в языке Фортран и float в C);
- ***с двойной точностью*** (соответствует типам REAL*8 в языке Фортран и double в C);
- ***с расширенной точностью*** (условно говоря, соответствует типам REAL*10 и более в языке Фортран и long double в C).

Число в представлении с простой точностью занимает 32 двоичных разряда: 23 разряда занимает мантисса и 8 разрядов отведено для порядка. Старший разряд является знаковым.

S	Мантисса	Порядок
1	23	8

Нормализованная форма чисел с плавающей точкой

Числа с плавающей точкой хранятся в *нормализованном виде*:

- в нормализованной форме точка расположена перед первой значащей, то есть, отличной от нуля, цифрой мантиссы;
- старший бит мантиссы всегда равен единице, он явным образом не указывается, а свободная позиция отводится под знак мантиссы. Таким образом при фиксированном количестве разрядов можно записать наибольшее количество значащих цифр и обеспечить наибольшую точность представления вещественного числа.

Мантисса нормализованного числа, если она не равна нулю, принадлежит диапазону [0.5, 1), в общем случае:

$$p^{-1}\left(1-\frac{1}{2}p^{-t}\right) \le m < 1-\frac{1}{2}p^{-t}$$

Порядок задается в формате с избытком (смещением) — истинное значение порядка увеличивается на 127, сумма всегда положительна. Фактическое значение порядка находится в промежутке от –126 до +127. Основанием является 2.
Младший бит мантиссы в формате с простой точностью представляет значение 2^{-24} (примерно 10^{-7}), что соответствует 7 значащим цифрам десятичного представления.
Значащие цифры числа допускают точное представление. Следующие значения имеют одинаковое (равное четырем) число значащих цифр: 3.142, 0.003142, 3.142e3.

В формате с простой точностью не имеет смысла хранить значения, содержащие более 8 десятичных разрядов мантиссы. Минимальное значение порядка -126 определяет минимальное по модулю, отличное от нуля, машинное число (около $1.17x10^{-38}$). Максимальное значение порядка составляет 127, что приблизительно соответствует значению $1.70x10^{38}$.

Число в представлении *с двойной точностью* занимает 64 двоичных разряда, из которых 52 разряда отводятся мантиссе и 11 разрядов порядку.

Для чисел с двойной точностью в десятичной системе диапазон значений составляет:
от $2.22x10^{-308}$ до $1.79x10^{308}$.

Количество значащих цифр и пределы изменения в этом случае больше, чем в формате с простой точностью (до 16 значащих цифр).

Расширенный формат используется для повышения точности промежуточных результатов вычислений.

Исходные данные для вычислений задаются с простой или двойной точностью, промежуточные вычисления выполняются с расширенной точностью, а окончательный результат формируется преобразованием к простой или двойной точности.

Расширенный формат используется и для вычисления значений математических функций. Диапазон значений от $3.4x10^{-4932}$ до $1.2x10^{4932}$.

Особые значения
Порядок 255 при нулевой мантиссе представляет значение «бесконечность». Порядок 255 при ненулевой мантиссе представляет значение, которое обозначается символьной строкой NaN (Not-a-Number, *не число*). Оно возникает при выполнении недопустимой операции вроде деления нулевого значения на нулевое или извлечения квадратного корня из отрицательного числа.

Если порядок равен нулю, а мантисса отлична от нуля, число называется *анормальным* (*субнормальным*, *денормализованным*). Оно может быть использовано для представления положительного и отрицательного нулей, а также значений, меньших минимального нормализованного числа. Субнормальные числа заполняют щель между нулем и наименьшим ненулевым нормализованным значением. Для чисел с простой точностью минимальное субнормальное значение равно $1.4x10^{-45}$, для чисел с двойной точностью $4.9x10^{-324}$, а для чисел с расширенной точностью не более $3.6x10^{-4951}$.
Положительный и отрицательный нули отличаются с точки зрения операции деления на ноль, которая в первом случае дает особое значение «положительная бесконечность», а во втором — «отрицательная бесконечность».

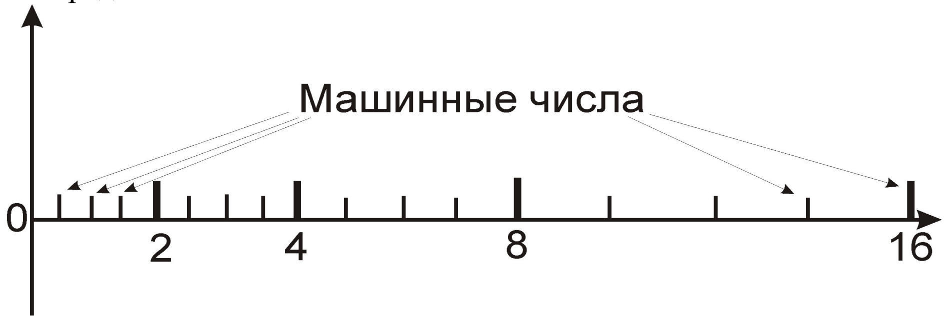
Существуют также форматы IBM, CRAY и некоторые другие. Они не получили широкого распространения.

Свойства множества машинных чисел

- *дискретность* — любые два машинных числа разделены на числовой оси конечным промежутком. В диапазоне [0, 1] из всех чисел с одинаковым количеством знаков после десятичной точки только пятая часть может быть точно представлена в двоичной форме;

- *конечность* — имеются наименьшее и наибольшее машинные числа;
- *неравномерное распределение на числовой оси* — расстояние между двумя соседними машинными числами вблизи нуля меньше, чем вблизи наибольшего или наименьшего значений. Объясняется неравномерность тем, что между двумя ближайшими степенями 2 всегда содержится одинаковое количество вещественных чисел, а поскольку расстояние между степенями возрастает с увеличением степени, возникает неравномерность в распределении машинных чисел.

Распределение машинных чисел на числовой оси:



Количественные характеристики множества машинных чисел:
- максимальное машинное число;
- минимальное машинное число;
- минимальное машинное число, отличное от нуля;
- машинное эпсилон.

Максимальное и минимальное машинные числа задают диапазон представимых чисел. В арифметике с плавающей точкой диапазон определяется количеством разрядов экспоненты.

Справедливы следующие оценки для абсолютной величины машинного числа:

$$2^{L-1} \le \left| x_{fl} \right| \le 2^{U}(1-2^{-t})$$

где L и U — минимальное и максимальное значения порядка, а t — разрядность мантиссы.

Количество машинных чисел:

$$N_{fl} = 2^{t}\left(U - L + 1\right) + 1$$

Наименьшее субнормальное число:

$$2^{L-t}$$

Машинное эпсилон представляет собой расстояние между единицей и ближайшим справа машинным числом:

$$\varepsilon_m = 2^{1-t}$$

Это — наименьшее из машинных чисел, для которого (в смысле машинной арифметики):

$$1 + \varepsilon_m > 1$$

Машинное эпсилон определяется размером мантиссы. Для простой точности его величина составляет приблизительно
$1.19 \text{x} 10^{-7}$
Его не следует путать с наименьшим вещественным числом, допускающим представление в заданном формате.

Машинные числа равномерно распределены на каждом из интервалов

$$\left[2^e, 2^{e+1}\right]$$

где e — порядок.

Расстояние между соседними машинными числами:

$$2^{e-t}$$

Увеличение порядка на единицу увеличивает длину интервала и, следовательно, удваивает это расстояние (в двоичном представлении).

Абсолютная погрешность машинного представления:

$$E(x) \le 2^{e-t-1}$$

Относительное расстояние между двумя последовательными машинными числами:

$$\left|\frac{\Delta x_{fl}}{x_{fl}}\right| = \frac{1}{m}$$