Конспекты лекций по курсу

«Введение в информатику и системы программирования», 1 семестр

С.А. Немнюгин, направление «Прикладные математика и физика»)

Лекция 10

Архитектура ЭВМ

Форматы хранения данных. Машинная арифметика

Отображение произвольных численных значений на множество машинных чисел

Особые ситуации при выполнении арифметических операций.
Даже если операнды арифметической операции являются машинными числами, ее результат может не принадлежать множеству машинных чисел. В этом случае возможны следующие ситуации:

- *переполнение* (overflow) - результат больше максимального или меньше минимального машинного числа. Переполнение приводит к завершению выполнения программы;
- *потеря точности* (потеря значимости, исчезновение порядка — underflow) - результат меньше наименьшего, отличного от нуля, машинного числа. *Чистый выход за пределы точности* (graceful underflow), при использовании субнормальных чисел;
- результат находится в промежутке между двумя машинными числами. В этом случае к результату применяется процедура *округления*.

Применение расширенного формата
В регистрах процессоров Интел для операций с плавающей запятой используется расширенный формат.
Любые значения, пересылаемые из памяти в любом формате, «на лету» преобразуются в расширенный формат. Значение при этом не изменяется.
Значения, пересылаемые из регистра в память с ячейками меньшей разрядности, округляются, и это может вызвать переполнение или потерю точности.

Искажение значений при вводе
Автоматическое преобразование из внешнего, десятичного представления, во внутренний, двоичный формат, производится при вводе дробных значений. Только целое значение может быть преобразовано в двоичное представление точно. Дробное число в общем случае может быть преобразовано во внутренний формат лишь приближенно.

Пример
Число 0.110, содержащее конечное число дробных разрядов в десятичном представлении, при переходе в двоичное представление, превращается в бесконечную периодическую двоичную дробь 0.00011001100…2. Такое число не может быть точно сохранено во внутреннем представлении, поэтому выполняется округление и замена данного значения ближайшим машинным числом.

Разряды защиты
Преобразование выполняется и при переходе от промежуточного представления в расширенном формате в выходной формат с простой или двойной точностью. При выполнении операций с плавающей точкой используются *разряды защиты* — дополнительные разряды для хранения промежуточных результатов.

Пример
Имеется компьютер, использующий десятичное представление с двумя значащими разрядами. Пусть выполняется операция:
1 - 0.99
Операнды хранятся в следующем виде: $0.1x10^1$ и $0.99x10^0$. При вычитании сначала выполняется выравнивание порядков, в результате получаем для уменьшаемого и вычитаемого: $0.1x10^1$ и $0.09x10^1$.

При выравнивании порядков в вычитаемом «потерялась» одна значащая цифра!!! Разность $0.01x10^1 = 0.1$, что на порядок отличается от правильного результата! Наличие дополнительного разряда для хранения промежуточного результата позволяет избежать большой погрешности.

Простое усечение
Существуют разные способы преобразования дробного значения к числу с заданной разрядностью. Один из способов заключается в отбрасывании «лишних» разрядов. Если исходное значение (в формате с простой точностью):

$$x = (.a_1 a_2 \ldots a_{24} a_{25} \ldots) \times 2^m$$

то машинное число, получаемое в результате *усечения*:

$$x' = (.a_1 a_2 \ldots a_{24}) \times 2^m$$

где $\tilde{a}_{24} = a_{24}$

При использовании простого усечения диапазон ошибок несимметричен относительно нуля. Это *смещенное приближение*. Если

$$2^{m-1} \le |x| < 2^m$$

то

$$x' = sign(x) \lfloor 2^{t-m} |x| \rfloor 2^{m-t}$$

где $\lfloor x \rfloor$ обозначает наибольшее целое, не превосходящее *x*.

Фоннеймановское усечение
Фоннеймановское усечение является *симметричным*. Оно формируется заменой младшего двоичного разряда результата единицей, если хотя бы один из отбрасываемых разрядов отличен от нуля. «Лишние» нулевые разряды просто отбрасываются.

При выполнении большого количества вычислений предпочтительным является симметричное фон-неймановское усечение.

Округление
Наибольшую точность обеспечивает ***округление***. Если старший из удаляемых двоичных разрядов равен 1, к младшему из остающихся двоичных разрядов прибавляется 1: $\tilde{a}_t = a_t$ для $a_{t+1} = 0$ и $\tilde{a}_t = a_t + 1$ для $a_{t+1} = 1$:

$$x''' = ((.a_1 a_2 \ldots a_{24}) + 2^{-24}) \times 2^m$$

Результат округления находится в промежутке между результатами двух типов усечения.

Усечение и округление
Техническая реализация округления является наиболее сложной. Округление обладает *свойством монотонности*: если для исходных чисел выполняется отношение $x < y$, то после

округления будет выполняться $\tilde{x}<\tilde{y}$ (символом «волна» здесь обозначены результаты округления).

Стандарты не определяют строго точность преобразования вещественных значений из десятичного представления в двоичное, поэтому они не гарантируют тождественность представления результатов в различных реализациях арифметики. Следует учитывать также, что при выполнении промежуточных операций могут использоваться различные типы промежуточных значений. Эти особенности могут привести к тому, что результат выполнения вычислительной программы на разных платформах будет разным.

Оценки погрешности представления произвольного вещественного числа для различных преобразований в формате с простой точностью:

$$|x - x'| \le \frac{1}{2}|x'' - x'| = \frac{1}{2} \times 2^{-24} \times 2^{m} = 2^{m-25}$$

$$|x - x''| \le \frac{1}{2}|x'' - x'| = 2^{m-25}$$

Здесь x - точное значение, x' - результат простого усечения, x'' - результат фоннеймановского усечения

Относительная погрешность при усечении:

$$\left|\frac{x - x'}{x}\right| \le \frac{2^{m-25}}{m \times 2^{m}} \le \frac{2^{-25}}{\frac{1}{2}} = 2^{-24}$$

Если

$$\delta = \frac{x - x'}{x}$$

то

$$x' = x(1+\delta)$$

Единичная ошибка округления

$$|\delta| \le 2^{-24}$$

единичная ошибка округления. Ее значение зависит от реализации машинной арифметики.

Результатом округления является замена исходного вещественного числа машинным:

$$x \Rightarrow x_{fl} = x(1+\delta) \qquad |\delta| \le \varepsilon$$

где

$$\varepsilon = \frac{1}{2} p^{1-n}$$

в случае округления и

$$\varepsilon = p^{1-n}$$

в случае усечения.

Погрешность арифметических операций

Арифметические операции с целыми числами выполняются абсолютно точно.
Действия с дробными числами выполняются с погрешностью.

Причины:
1. Результат операции может оказаться не машинным числом.
2. Особенности выполнения операций сложения и вычитания. Перед выполнением сложения или вычитания порядки операндов выравниваются (приводятся к одинаковым значениям), мантиссы при этом сдвигаются друг относительно друга. Мантисса числа с меньшим порядком сдвигается вправо на количество разрядов, равное разности порядков. При этом может происходить потеря значащих разрядов. Порядок результата устанавливается равным большему из порядков. Результат операции в случае необходимости нормализуется.

При умножении или делении чисел выравнивание порядков не выполняется.

Может оказаться так, что в процессе вычислений возникает:
1. слишком маленькое (порядок меньше –126 или –1022);
2. слишком большое (порядок превышает +127 или +1023) значение.

В обоих случаях значения не могут быть представлены в стандартном формате. В первом случае говорят о *потере значимости*, во втором — о *переполнении разрядной сетки*.

Это — особые ситуации, называемые *арифметическими исключениями*.

Пусть $\otimes$ — произвольная арифметическая операция, применяемая к истинным вещественным числам, а

$\tilde{\otimes}$ — ее машинный аналог, тогда:

$$x \,\tilde{\otimes}\, y = (x \otimes y)(1+\delta) \qquad \text{где} \qquad \delta \le \frac{1}{2}\varepsilon_m$$

Этому условию удовлетворяют только операции, в реализации которых используются разряды защиты

Можно показать, что для операции суммирования относительная погрешность:

$$\frac{\left|x \,\tilde{\oplus}\, y - (x+y)\right|}{|x+y|} \le \varepsilon(1+\varepsilon)\frac{|x|+|y|}{|x+y|} + \varepsilon$$

Отсюда следует, что относительная погрешность мала, если сумма достаточно велика. Если же слагаемые имеют противоположные знаки и близки по абсолютной величине, знаменатель в правой части оказывается маленьким и относительная погрешность возрастает. В этом случае возникает *погрешность сокращения*. Избежать ее можно, используя, альтернативные формулы.

Пример
Среднее арифметическое двух значений можно вычислить по любой из следующих формул:

$$m = \frac{x+y}{2}$$

и

$$m = x + \frac{y-x}{2}$$

Первый вариант оказывается более точным, если оба значения имеют одинаковые знаки и второй — если противоположные.

Величина погрешности вычисления разности дается следующей теоремой.

ТЕОРЕМА

Если x и y - положительные нормализованные числа с плавающей точкой в двоичном представлении, $x>y$ и $2^{-q} \le 1-\frac{y}{x} \le 2^{-p}$.

Тогда при вычислении разности $x-y$ теряется от p до q значащих цифр.

Следствием погрешности представления вещественных чисел и округлений является утеря некоторых свойств арифметических операций.
При переходе к машинной арифметике сохраняются коммутативность (перестановочное свойство) сложения и умножения.
Ассоциативность (сочетательное свойство) этих операций нарушается.

Форматы хранения символьной информации

Код **ASCII** (**A**merican **S**tandard **C**ode for **I**nformation **I**nterchange.

Каждому символу сопоставляется 7- или 8-битный (1-байтовый) числовой код.

Расширенная таблица ASCII использует 8 двоичных разрядов и состоит из двух частей. Первая, в которую входят символы с кодами 0–127, является универсальной, а вторая (коды 128–255) предназначена для специальных символов и букв национальных алфавитов (в том числе и русского). Символы в первых позициях этой таблицы являются управляющими.

Символы ASCII с кодами 0–127

Код	Символ	Код	Символ	Код	Символ	Код	Символ
0	NUL	32	SP	64	@	96	`
1	SOH	33	!	65	A	97	a
2	STX	34	"	66	B	98	b
3	ETX	35	#	67	C	99	c
4	EOT	36	$	68	D	100	d
5	ENQ	37	%	69	E	101	e
6	ACK	38	&	70	F	102	f
7	BEL	39	'	71	G	103	g
8	BS	40	(	72	H	104	h
9	HT	41	)	73	I	105	i
10	LF	42	*	74	J	106	j
11	VT	43	+	75	K	107	k
12	FF	44	,	76	L	108	l
13	CR	45	-	77	M	109	m
14	SO	46	.	78	N	110	n
15	SI	47	/	79	O	111	o
16	DLE	48	0	80	P	112	p
17	DC1	49	1	81	Q	113	q
18	DC2	50	2	82	R	114	r
19	DC3	51	3	83	S	115	s
20	DC4	52	4	84	T	116	t
21	NAK	53	5	85	U	117	u
22	SYN	54	6	86	V	118	v
23	ETB	55	7	87	W	119	w
24	CAN	56	8	88	X	120	x
25	EM	57	9	89	Y	121	y
26	SUB	58	:	90	Z	122	z
27	ESC	59	;	91	[	123	{
28	FS	60	<	92	\	124	|
29	GS	61	=	93	]	125	}
30	RS	62	>	94	^	126	~
31	US	63	?	95	_	127	DEL