

Лабораторна робота № 3

ВНУТРІШНЄ ПРЕДСАВЛЕННЯ ДІЙСНИХ ДАНИХ

Мета заняття: ознайомитися з дійсними числами; розглянути алгоритм переведення дійсних чисел з десяткової в двійкову системи числення і їх представлення в нормалізованому вигляді.

Теоретичні відомості

Поняття дійсних чисел. Їх формат і представлення.

Дійсні числа – це числа, які можуть містити у собі як цілі, так і дробові значення з точкою відокремлення від цілої частини. Для від'ємного числа треба ставити знак мінус (-) перед значенням (числом). Приклади дійсних чисел:

20.0005

99.9

–5000.12

–9999.9991

Дійсні базові величини можуть бути формату dword, qword або tbyte. Внутрішнє (машинне) перетворення цих чисел досить важке.

Формат дійсного числа

<i>S</i>	<i>Характеристика</i>	<i>Нормалізована мантиса</i>
----------	-----------------------	------------------------------

Біт **S** – знак числа.

Характеристика = Зміщення ± Порядок.

Зміщення – число, яке дорівнює половині максимального, яке може поміститися в поле *Характеристика*. Таким чином, економляться місце і час, так як не потрібно виділяти розряд для знака порядку і робити додатковий код для від'ємних порядків.

Розмір простору, який ПК використовує для збереження *Характеристики* і *Мантиси*, встановлений стандартом IEEE – 1985 і підтримується всіма сучасними комп'ютерними архітектурами.

Для отримання внутрішнього представлення дійсного числа потрібно в будь-якому випадку перевести його в двійкову форму.

Переведення дійсних чисел з десяткової в двійкову систему числення

Для переведення дійсного десяткового числа в двійкову систему числення необхідно дане число (тільки мантису) послідовно множити на число 2 до тих пір, поки в мантисі не вийде або чистий нуль, або необхідна кількість розрядів.

При множенні отримані цілі значення не враховуються. Зате вони послідовно зараховуються в результат.

Отримуємо впорядковану послідовність цілих двійкових чисел.

Приклад: перевести дійсні числа в двійкову систему числення.

Спочатку переведемо ціле число 105 в двійкову систему числення.

105	2					
104	52	2				
1	52	26	2			
	0	26	13	2		
		0	12	6	2	
			1	6	3	2
				0	2	1
					1	

105d → 1101001b

Тепер переведемо в двійкову систему числення правильний десятковий дріб. Для перевірки представлення в 32-х бітовому форматі нам досить 24-х біт в мантисі (зробимо із запасом – 26 біт)

0.4612. → ??? b

№ біта	Біт	Мантиса (Dec)	Множник	Пояснення
	0	4612	2	Початкове число
1	0	9224		Результати множення мантиси на 2
2	1	8448		
3	1	6896		
4	1	3792		
5	0	7584		
6	1	5168		
7	1	0336		
8	0	0672		
9	0	1344		
10	0	2688		
11	0	5376		
12	1	0752		

13	0	1504		
14	0	3008		
15	0	6016		
16	1	2032		
17	0	4064		
18	0	8128		
19	1	6256		
20	1	2512		
21	0	5024		
22	1	0048		
23	0	0096		
24	0	0192		
25	0	0384		
26	0	0768		

0.4612d → 0.01110110000100010011010000b

Тепер є вся інформація для отримання двійкового представлення змішаного дробу:

105.4612d → 1101001.01110110000100010011010000b

Представлення дійсних чисел в двійковому нормалізованому вигляді

Отримавши двійкове представлення, необхідно його нормалізувати, тобто двійкове число має завжди починатися з одиниці і мати наступний вигляд:

$$\pm 1. m_2 * 2^p,$$

де m_2 – двійкова мантия числа,
 p – порядок двійкового числа.

Приклад: представити двійкове число в нормалізованому вигляді:

$$\pm 1.0d \rightarrow \pm 1.0b * 2^0.$$

Приклад: представити десяткове число в нормалізованому вигляді:

$$\pm 0.5d \rightarrow \pm 0.1b,$$

в нормалізованому вигляді: $\pm 0.1b \rightarrow \pm 1.0b * 2^{-1}$.

Приклад: представити десяткове число в нормалізованому вигляді:

$$\pm 0.703125d \rightarrow \pm 0.101101b,$$

в нормалізованому вигляді: $\pm 0.101101b \rightarrow \pm 1.01101b * 2^{-1}$.

Приклад: представити десяткове число в нормалізованому вигляді:

$$\pm 0.05d \rightarrow \pm 0.000011(0011)b,$$

в нормалізованому вигляді: $\pm 0.000011(0011)b \rightarrow \pm 1.1(0011)b * 2^{-5}$.

Приклад: представити десяткове число в нормалізованому вигляді:

$$\pm 117.25d \rightarrow \pm 1110101.01b,$$

в нормалізованому вигляді: $\pm 1110101.01b \rightarrow \pm 1.11010101b * 2^6$.

Машинні формати дійсних чисел

Дійсні базові величини можуть бути формату dword, qword або tbyte. Всі вони мають для збереження комірки різної довжини. Звідси і різний діапазон представлення для різних типів дійсних чисел, хоча ідея збереження майже одна і та ж.

Формат Dword

Це число зберігається в комірці довжиною 32 біти, які розподіляються наступним чином:

1 – прихований розряд
V

S	Характеристика			Нормалізована мантиса				
31	30	...	23	22	21	...	1	0
Біти								

На *Характеристику* виділяється 8 бітів (розряди 23-30). Максимально можливе шістнадцяткове число, яке можна розмістити в цих бітах, дорівнює **FFh**, а його половина – це *Зміщення* = **7Fh**. Оскільки на представлення мантиси залишається всього 23 біти (розряди 0-22), а мантиса завжди є нормалізованою, то перша одиниця в інформаційні розряди мантиси не потрапляє (прихований розряд), а на мантису в результаті відводиться не 23 біти, а 24. Таке представлення дозволяє правильно відображати 7-8 десяткових цифр, а його діапазон представлення складає $1.5 * 10^{-45} \dots 3.4 * 10^{38}$.

Приклад: представити десяткове число в нормалізованому вигляді і подати його в 32-бітному форматі:

$$\pm 1.0d \rightarrow \pm 1.0 * 2^0b.$$

$$\text{Характеристика} = 7F + 0 = 7F.$$

Розписуємо детально склад 32-х бітів, а потім переводимо двійкове представлення в шістнадцяткове.

1 – прихований розряд
V

0	0	1	1	1	1	1	1	1	0	0	0	0	...	0	0
S	Характеристика								Нормалізована мантиса						
31	30	29	28	27	26	25	24	23	22	21	20	19	...	1	0
Біти															
3				F				8				0		0 0 0	
HEX-код															

Від’ємне число -1.0 буде таке ж саме, тільки в знаковому біті S=1:

1 – прихований розряд
V

1	0	1	1	1	1	1	1	1	0	0	0	0	...	0	0
S	Характеристика								Нормалізована мантиса						
31	30	29	28	27	26	25	24	23	22	21	20	19	...	1	0
Біти															
B				F				8				0		0 0 0	
HEX-код															

Приклад: представити десяткове число в нормалізованому вигляді і подати його в 32-бітному форматі:

$$\pm 0.5d \rightarrow \pm 1.0 * 2^{-1}b.$$

$$\text{Характеристика} = 7F - 1 = 7E.$$

Знову розписуємо склад 32-х бітів, а потім переводимо двійкове представлення в шістнадцяткове. В цьому прикладі змінився тільки порядок. В мантисі знову тільки один розряд, та й той прихований.

Додатне число 0.5:

1 – прихований розряд

v

0	0	1	1	1	1	1	1	0	0	0	0	...	0	0	
S	Характеристика								Нормалізована мантиса						
31	30	29	28	27	26	25	24	23	22	21	20	19	...	1	0
Біти															
3				F				0				0		0 0 0	
HEX-код															

Від’ємне число -1.0 буде таке ж саме, тільки в знаковому біті S=1:

1 – прихований розряд

v

1	0	1	1	1	1	1	1	0	0	0	0	...	0	0	
S	Характеристика								Нормалізована мантиса						
31	30	29	28	27	26	25	24	23	22	21	20	19	...	1	0
Біти															
B				F				0				0		0 0 0	
HEX-код															

Формат Qword

В цьому форматі ідея та ж сама, що і в попередньому. На все число відводиться комірка довжиною 64 біти. На характеристику виділено 11 бітів (розряди 52-62).

1 – прихований розряд
v

S	Характеристика			Нормалізована мантиса				
63	62	...	52	51	50	...	1	0
Біти								

Максимально можливе число, яке можна розмістити в 11-ти бітах: **111 1111 1111b**. Це означає, що:

Зміщення = 011 1111 1111b → 3FFh.

Мантиса також має прихований розряд. Таке представлення дозволяє отримати наступний допустимий діапазон для даних типу double:

$$5.0 * 10^{-324} \dots 1.7 * 10^{308},$$

і, відповідно, представити 15-16 десяткових цифр.

Приклад: представити десяткове число в нормалізованому вигляді і подати його в 64-бітному форматі:

$$\pm 0.05d \rightarrow \pm 1.1(0011) * 2^{-5}b.$$

$$\text{Характеристика} = 3FF - 5 = 3FA.$$

Розпишемо наше число спочатку в двійковому вигляді:

0 011 1111 1010 **1**001 1001 1001 1001 1001 ... 1001

Λ

1

Тепер можна представити його і в шістнадцятковому вигляді: **3FA999999999999h.**

Формат Tbyte

S	Характеристика			Нормалізована мантиса				
79	78	...	64	63	62	...	1	0
Біти								

На все число відводиться 80 бітів. Цей формат є основним робочим форматом для даних сопроцесора. Тому, щоб НЕ створювати додаткових перетворень при обчисленнях, у **мантиси нема прихованого розряду**. На характеристику витрачається 15 бітів (розряди 64-78). Максимально можливе число, яке можна розмістити в 15-ти бітах: **111 1111 1111 1111b**. Це означає, що:

Зміщення = 011 1111 1111 1111b → 3FFFh.

Таке представлення дозволяє отримати наступний допустимий діапазон для даних цього типу:

$$5.0 * 10^{-324} \dots 1.7 * 10^{308},$$

і, відповідно, представити 19-20 десяткових цифр.

Приклад: представити десяткове число в нормалізованому вигляді і подати його в 80-бітному форматі:

$$\pm 117.25d \rightarrow \pm 1.11010101 * 2^6b.$$

Характеристика = 3FFF + 6 = 4005 (при додаванні повна аналогія з десятковою системою числення).

Розпишемо наше число спочатку в двійковому вигляді:

0 100 0000 0000 0101 1110 1010 1000 0000 ...
0000b

$$117.25d \rightarrow 4005EA80000000000000h$$

1 100 0000 0000 0101 1110 1010 1000 0000 ...
0000b

$$-117.25d \rightarrow C005EA80000000000000h$$

Реалізація в turbo assembler 3.2

```
.MODEL tiny
.DATA

;-----float (DWord)
f0 dd -105.4612
   dd 105.4612
   dd -15.4522
   dd 15.4522
   dd 105.
   dd -105.
   dd 15.
   dd -15.
   dd 0.4612
   dd -0.4612
   dd 0.4522
   dd -0.4522
;----- double (QWord)
f1 dq -105.4612
   dq 105.4612
   dq -15.4522
   dq 15.4522
   dq 105.
   dq -105.
   dq 15.
   dq -15.
   dq 0.4612
   dq -0.4612
   dq 0.4522
   dq -0.4522
;----- long double
f2 dt -105.4612
   dt 105.4612
   dt -15.4522
   dt 15.4522
   dt 105.
   dt -105.
   dt 15.
   dt -15.
   dt 0.4612
   dt -0.4612
   dt 0.4522
   dt -0.4522

END
```

```

1 0000 .MODEL tiny
2 0000 .DATA
3 ;-----float (DWord)
4 0000 C2D2EC22 f0 dd -105.4612
5 0004 42D2EC22 dd 105.4612
6 0008 C1773C36 dd -15.4522
7 000C 41773C36 dd 15.4522
8 0010 42D20000 dd 105.
9 0014 C2D20000 dd -105.
10 0018 41700000 dd 15.
11 001C C1700000 dd -15.
12 0020 3EEC2268 dd 0.4612
13 0024 BEEC2268 dd -0.4612
14 0028 3EE786C2 dd 0.4522
15 002C BEE786C2 dd -0.4522
16 ;-----double (QWord)
17 0030 C05A5D844D013A92 f1 dq -105.4612
18 0038 405A5D844D013A92 dq 105.4612
19 0040 C02EE786C226809D dq -15.4522
20 0048 402EE786C226809D dq 15.4522
21 0050 405A400000000000 dq 105.
22 0058 C05A400000000000 dq -105.
23 0060 402E000000000000 dq 15.
24 0068 C02E000000000000 dq -15.
25 0070 3FDD844D013A92A3 dq 0.4612
26 0078 BFD0844D013A92A3 dq -0.4612
27 0080 3FDCF0D844D013A9 dq 0.4522
28 0088 BFD0CF0D844D013A9 dq -0.4522
29 ;-----long double
30 0090 C005D2EC226809D49518 f2 dt -105.4612
31 009A 4005D2EC226809D49518 dt 105.4612
32 00A4 C002F73C36113404EA4A dt -15.4522
33 00AE 4002F73C36113404EA4A dt 15.4522
34 00B8 4005D2000000000000 dt 105.
35 00C2 C005D2000000000000 dt -105.
36 00CC 4002F0000000000000 dt 15.
37 00D6 C002F0000000000000 dt -15.
38 00E0 3FFDEC226809D495182A dt 0.4612
39 00EA BFFDEC226809D495182A dt -0.4612
40 00F4 3FFDE786C226809D4951 dt 0.4522
41 00FE BFFDE786C226809D4951 dt -0.4522
42 END
    
```

Завдання на лабораторну роботу

1. Для заданих чисел (табл.3.1) виконати перетворення у внутрішній (машинний) формат подання чисел в пам'яті комп'ютера за умови, що числа є знаковими. Для збереження чисел використовується 32 біти.

Таблиця 3.1

**Дані для перетворення знакових дійсних чисел
у внутрішній машинний формат**

№ варіанта	Число 1	Число 2	Число 3
1	±3573,00	±0,00092	±3573,00092
2	±3811,00	±0,00124	±3811,00124
3	±4707,00	±0,00292	±4707,00292
4	±3726,00	±0,00077	±3726,00077
5	±3492,00	±0,00073	±3492,00073
6	±4448,00	±0,00301	±4448,00301
7	±3157,00	±0,00024	±3157,00024
8	±3098,00	±0,00061	±3098,00061
9	±4535,00	±0,00252	±4535,00252
10	±3309,00	±0,00012	±3309,00012
11	±3968,00	±0,00095	±3968,00095
12	±4322,00	±0,00183	±4322,00183
13	±4853,00	±0,00364	±4853,00364
14	±3288,00	±0,00013	±3288,00013
15	±3675,00	±0,00152	±3675,00152
16	±4206,00	±0,00271	±4206,00271
17	±3799,00	±0,00084	±3799,00084
18	±4337,00	±0,00312	±4337,00312
19	±3862,00	±0,00099	±3862,00099
20	±4571,00	±0,00268	±4571,00268
21	±3444,00	±0,00015	±3444,00015
22	±4123,00	±0,00298	±4123,00298

Продовження табл.3.1

**Дані для перетворення знакових дійсних чисел
у внутрішній машинний формат**

№ варіанта	Число 1	Число 2	Число 3
23	±3631,00	±0,00022	±3631,00022
24	±4486,00	±0,00233	±4486,00233
25	±4662,00	±0,00315	±4662,00315
26	±3925,00	±0,00198	±3925,00198
27	±4243,00	±0,00342	±4243,00342
28	±4018,00	±0,00179	±4018,00179
29	±4955,00	±0,00236	±4955,00236
30	±4166,00	±0,00323	±4166,00323
31	±3264,00	±0,00421	±3264,00421
32	±4026,00	±0,00189	±4026,00189
33	±3765,00	±0,00056	±3765,00056
34	±3242,00	±0,00166	±3242,00166
35	±4661,00	±0,00345	±4661, 00345

2. Для заданих чисел (табл.3.1) виконати перетворення у внутрішній(машинний) формат подання чисел в пам'яті комп'ютера за умови, що числа є знаковими. Для збереження чисел використовується 64 біти.

3. Для заданих чисел (табл.3.1) виконати перетворення у внутрішній(машинний) формат подання чисел в пам'яті комп'ютера за умови, що числа є знаковими. Для збереження чисел використовується 80 біт.

4. Для отриманих результатів в завданні 1-3 здійснити перевірку за допомогою turbo assembler.

Контрольні питання

1. Що таке дійсні числа?
2. Наведіть приклади дійсних чисел.
3. Формат представлення дійсних чисел.
4. Що таке зміщення?
5. Алгоритм переведення дійсних чисел з десяткової в двійкову систему числення.
6. Алгоритм представлення дійсних чисел в двійковому нормалізованому вигляді.
7. Які існують машинні формати дійсних чисел?
8. Наведіть значення зміщень для кожного машинного формату дійсних чисел.
9. Формати машинного представлення дійсних чисел Dword і Qword.
10. Формат машинного представлення дійсних чисел Tbyte.