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 цисел досить важке.

Формат дійсного числа

S	*Характеристика*	*Нормалізована мантиса*

Біт **S** – знак числа.

Характеристика = Зміщення ± Порядок.

Зміщення – число, яке дорівнює половині максимально можливого, яке може поміститися в поле *Характеристика.* Таким чином, економляться місце і час, так як не потрібно виділяти розряд для знака порядка 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

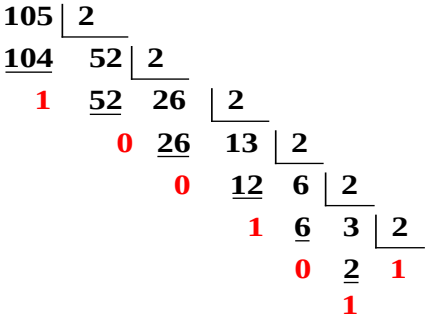
Для переведення дійсного десяткового числа в двійкову систему числення необхідно дане число (тільки мантису) послідовно множити на число 2 до тих пір, поки в мантисі не вийде або чистий нуль, або необхідна кількість розрядів.

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

Отримуємо впорядковану послідовність цілих двійкових чисел.

Приклад: перевести дійсне число в двійкову систему числення.

Спочатку переведемо ціле число 105 в двійкову систему числення.



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 d → ??? b

№ біта	Біт	Мантиса (Dec)	Множник	Пояснення
	0	0,4612	2	Початкове число
1	0	0,9224		Результати множення дробової мантиси на 2
2	1	1,8448		
3	1	1,6896		
4	1	1,3792		
5	0	0,7584		
6	1	1,5168		
7	1	1,0336		
8	0	0,0672		
9	0	0,1344		
10	0	0,2688		
11	0	0,5376		
12	1	1,0752		
13	0	0,1504		
14	0	0,3008		
15	0	0,6016		
16	1	1,2032		
17	0	0,4064		
18	0	0,8128		
19	1	0,6256		
20	1	0,2512		
21	0	0,5024		
22	1	1,0048		
23	0	0,0096		
24	0	0,0192		
25	0	0,0384		
26	0	0,0768		

0.4612 d → 0. 01110110000100010011010000 b

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26

Номери бітів (розрядів) →

Тепер є вся інформація для отримання двійкового представлення змішаного дробу:

105.4612 d → 1101001.01110110000100010011010000 b

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 мантиса 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* = 7**Fh.** 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 $1.5 * 10^{-45}$.. $3.4 * 10^{38}$.

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

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

Характеристика = 7F + 0 = 7F.

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

Додатне число 1.0:

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:

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

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	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	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:

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

і, відповідно, представити 15-16 десяткових цифр.

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

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

Характеристика = 3FF – 5 = 3FA.

Запишемо наше додатне число спочатку в двійковому вигляді:

$$\underset{\textit{знак}}{\mathbf{0}} \quad \underbrace{\mathbf{011\ 1111\ 1010}}_{\textit{характеристика}} \underbrace{\mathbf{1}\mathbf{0011001\ 1001\ 1001\ 1001\ \ldots\ 1001}}_{\textit{мантиса}}$$

$1 - \textit{прихований розряд}$

Тепер можна представити його і в шістнадцятковому вигляді:

3FA9999999999999h.

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:

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

і, відповідно, представити 19-20 десяткових цифр.

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

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

$$Характеристика = 3FFF + 6 = 4005$$

(при додаванні повна аналогія з десятковою системою числення).
Запишемо наше додатне число спочатку в двійковому вигляді:

відсутній прихований розряд

0 **100 0000 0000 0101** **1110 1010 1000 0000 … 0000**b

знак *характеристика* *мантиса*

В шістнадцятковому вигляді: $\mathbf{117.25}\boldsymbol{d} \rightarrow \mathbf{4005}\boldsymbol{EA}\mathbf{80000000000000}\boldsymbol{h}$

Від'ємне число в двійковому вигляді:

відсутній прихований розряд

1 **100 0000 0000 0101** **1110 1010 1000 0000 … 0000**b

знак *характеристика* *мантиса*

В шістнадцятковому вигляді: $\mathbf{-117.25}\boldsymbol{d} \rightarrow \mathbf{C005}\boldsymbol{EA}\mathbf{80000000000000}\boldsymbol{h}$

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
```

Файл лістингу lr_03.lst

```
Turbo Assembler  Version 5.0        10-09-22 18:22:10       Page 1
lr_03.asm
|


 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  BFDD844D013A92A3                  dQ  -0.4612
27 0080  3FDCF0D844D013A9                  dQ   0.4522
28 0088  BFDCF0D844D013A9                  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  4005D200000000000000              dT   105.
35 00C2  C005D200000000000000              dT  -105.
36 00CC  4002F000000000000000              dT    15.
37 00D6  C002F000000000000000              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
```

Правила формування завдань

Якщо унікальний номер групи **G** = **1**, то значення для 1 завдання згідно таблиці 3.1 будуть мати вигляд:

№ варіанта (N)	Число 1 (ціла частина)	Число 2 (дробова частина)	Число 3 (ціла + дробова частини)
1	±3573+**G**,00	±0,00092+**G**	±3573+**G**,00092+**G**

1 число = ±3573+**G**,00 = ±3573+**1**,00 = ±**3574**,**00**

2 число = ±0,00092+**G** =±0,00092+**1** = ±**0,00093**

3 число = ±3573+**G**,00092+**G** =±3573+**1**,00092+**1** = ±**3574,00093**

Всі інші варіанти формуються аналогічно!!!

Вимоги до оформлення звіту

Звіт оформлюється у текстовому документі і має містити:

1. Правильно оформлену рамку.
2. Тему, мету, номер варіанту.
3. Виконані **всі** завдання з ходом обрахунків.
4. Висновки.

Звіт оформлений неналежним чином і без ходу обрахунків не оцінюється і має бути перероблений!

Завдання на лабораторну роботу

1. Для заданого числа (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. Для отриманих результатів в завданні здійснити перевірку за допомогою turbo assembler.

Таблиця 3.1

Дані для перетворення знакових дійсних чисел у внутрішній машинний формат

G – унікальний номер групи зазначений на сторінці курсу.

Варіант №	Число 1	Варіант №	Число 1
1	±5274+**G**,00	**19**	±5694+**G**,00
2	±4183+**G**,00	**20**	±4938+**G**,00
3	±5941+**G**,00	**21**	±4265+**G**,00
4	±4637+**G**,00	**22**	±5503+**G**,00
5	±5419+**G**,00	**23**	±4619+**G**,00
6	±4076+**G**,00	**24**	±5874+**G**,00
7	±5782+**G**,00	**25**	±5058+**G**,00
8	±4358+**G**,00	**26**	±4147+**G**,00
9	±5126+**G**,00	**27**	±5326+**G**,00
10	±4893+**G**,00	**28**	±5769+**G**,00
11	±5567+**G**,00	**29**	±4402+**G**,00
12	±4241+**G**,00	**30**	±5486+**G**,00
13	±5836+**G**,00	**31**	±4961+**G**,00
14	±4715+**G**,00	**32**	±4219+**G**,00
15	±5392+**G**,00	**33**	±5638+**G**,00
16	±4028+**G**,00	**34**	±4527+**G**,00
17	±5187+**G**,00	**35**	±5805+**G**,00
18	±4473+**G**,00	**-**	

2. Для заданих чисел (табл.3.2) 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. Для отриманих результатів в завданні здійснити перевірку за допомогою turbo assembler.

Таблиця 3.2

Дані для перетворення знакових дійсних чисел у внутрішній машинний формат

G – унікальний номер групи зазначений на сторінці курсу.

Варіант №	Число 2	Варіант №	Число 2
1	±0,00347+**G**	**19**	±0,00193+**G**
2	±0,00116+**G**	**20**	±0,00358+**G**
3	±0,00283+**G**	**21**	±0,00051+**G**
4	±0,00058+**G**	**22**	±0,00227+**G**
5	±0,00391+**G**	**23**	±0,00384+**G**
6	±0,00172+**G**	**24**	±0,00137+**G**
7	±0,00214+**G**	**25**	±0,00294+**G**
8	±0,00043+**G**	**26**	±0,00074+**G**
9	±0,00328+**G**	**27**	±0,00316+**G**
10	±0,00105+**G**	**28**	±0,00181+**G**
11	±0,00267+**G**	**29**	±0,00256+**G**
12	±0,00373+**G**	**30**	±0,00342+**G**
13	±0,00086+**G**	**31**	±0,00047+**G**
14	±0,00239+**G**	**32**	±0,00169+**G**
15	±0,00148+**G**	**33**	±0,00367+**G**
16	±0,00305+**G**	**34**	±0,00203+**G**
17	±0,00069+**G**	**35**	±0,00129+**G**
18	±0,00276+**G**	**-**	

3. Для заданих чисел (табл.3.3) 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. Для отриманих результатів в завданні здійснити перевірку за допомогою turbo assembler.

Таблиця 3.3

Дані для перетворення знакових дійсних чисел у внутрішній машинний формат

G – унікальний номер групи зазначений на сторінці курсу.

Варіант №	Число 3	Варіант №	Число 3
1	±2210+**G**,00329+**G**	**19**	±2874+**G**,00465+**G**
2	±2002+**G**,00264+**G**	**20**	±3538+**G**,00473+**G**
3	±3803+**G**,00428+**G**	**21**	±3155+**G**,00478+**G**
4	±3714+**G**,00479+**G**	**22**	±3932+**G**,00237+**G**
5	±3427+**G**,00354+**G**	**23**	±2081+**G**,00497+**G**
6	±3693+**G**,00525+**G**	**24**	±2211+**G**,00331+**G**
7	±3652+**G**,00501+**G**	**25**	±2844+**G**,00333+**G**
8	±3702+**G**,00302+**G**	**26**	±2843+**G**,00472+**G**
9	±2282+**G**,00568+**G**	**27**	±2396+**G**,00310+**G**
10	±2840+**G**,00413+**G**	**28**	±2386+**G**,00453+**G**
11	±3550+**G**,00533+**G**	**29**	±3984+**G**,00548+**G**
12	±3250+**G**,00293+**G**	**30**	±3776+**G**,00423+**G**
13	±2638+**G**,00208+**G**	**31**	±2651+**G**,00541+**G**
14	±3353+**G**,00280+**G**	**32**	±2343+**G**,00511+**G**
15	±2604+**G**,00323+**G**	**33**	±3634+**G**,00459+**G**
16	±3645+**G**,00425+**G**	**34**	±3407+**G**,00328+**G**
17	±2433+**G**,00494+**G**	**35**	±2870+**G**,00439+**G**
18	±2554+**G**,00403+**G**	**-**	

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.