**Міністерство освіти і науки України**

**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Факультет інформатики та обчислювальної техніки**

**Кафедра обчислювальної техніки**

Лабораторна робота №2

з дисципліни

«Системне програмування»

на тему

«Внутрішнє представлення цілих і дійсних даних у процесорі ix86»

Виконала: Перевірив:

Студентка групи ІМ-21 доцент Павлов В. Г.

Дворецька Анастасія Вадимівна

Номер у списку групи: 6

**Київ 2024**

Варіант 6

**Мета:**

Вивчення форматів внутрішнього представлення цілих і дійсних чисел шляхом порівняння результатів, отриманих теоретично, з результатами, що містяться в отриманому розширеному лістингу програми.

**1.Cформовані початкові числа на основі дати народження,записаної у форматі dd.mm.yyyy(16.05.2005):**

N = 8760;

А = 16 ;

B = 1605;

C = 16052005;

D = A/N = 16/8760 = 0,00182648 = 0,002;

E = B/N = 1605/8760 = 0,18321918 = 0,183;

F = C/N = 16052005/8760 = 1832,42066 = 1832,421.

**2.Отримані числа в двійковій системі числення (дробову частину округлила до 10 знаків після коми):**

**A** = 16 => 10000

**B** = 1605 => 11001000101

**C** = 16052005 => 1111 0100 1110 1111 0010 0101

**D** = 0,002 => 0,0000000010

Ціла частина:

0 => 0

Дробова:

0,002 \* 2 = **0**,004

0,004 \* 2 = **0**,008

0,008 \* 2 = **0**,016

0,016 \* 2 = **0**,032

0,032 \* 2 = **0**,064

0,064 \* 2 = **0**,128

0,128 \* 2 = **0**,256

0,256 \* 2 = **0**,512

0,512 \* 2 = **1**,024

0,024 \* 2 = **0**,048

0,048 \* 2 = 0,096

**E** = 0,183 => 0,0010111011

Ціла частина:

0 => 0

Дробова:

0,183 \* 2 = **0**,366

0,366 \* 2 = **0**,732

0,732 \* 2 = **1**,464

0,464 \* 2 = **0**,928

0,928 \* 2 = **1**,856

0,856 \* 2 = **1**,712

0,712 \* 2 = **1**,424

0,424 \* 2 = **0**,848

0,848 \* 2 = **1**,696

0,696 \* 2 = **1**,392

0,392 \* 2 = 0,784

**F** = 1832,421 => 11100101000,0110101111

Ціла частина:

1832 => 11100101000

Дробова:

0,421 \* 2 = **0**,842

0,842 \* 2 = **1**,684

0,684 \* 2 = **1**,368

0,368 \* 2 = **0**,736

0,736 \* 2 = **1**,472

0,472 \* 2 = **0**,944

0,944 \* 2 = **1**,888

0,888 \* 2 = **1**,776

0,776 \* 2 = **1**,552

0,552 \* 2 = **1**,104

0,104 \* 2 = 0,208

**3.За допомогою розрядної сітки показати представлення цілих чисел в наступних форматах:**

**1) “ddmmyyyy” у вигляді символьного рядка;**

1 -> 3116

6 -> 3616

0 -> 3016

5 -> 3516

2 -> 3216

0 -> 3016

0 -> 3016

5 -> 3516

**2) числа A и –A у однобайтовому форматі Byte;**

| 0001 | 0000 |
| --- | --- |

**А** = 1610 = 0001 00002 = 1016,

Щоб знайти -А,яке дорівнює -16,потрібно:

1. Взяти двійковий код додатнього А:

| 0001 | 0000 |
| --- | --- |

1. Інвертувати кожен біт(0 -> 1, 1->0):

| 1110 | 1111 |
| --- | --- |

1. Додати 1: 1111

| 1111 | 0000 |
| --- | --- |

Отже, **-А** = -1610 = 1111 00002 = F016

**3) числа A, B, -A и -B у двобайтовому формат Word;**

**А** = 1610 = 0000 0000 0001 00002 = 00 1016

| 0000 | 0000 | 0001 | 0000 |
| --- | --- | --- | --- |

**В** = 160510 = 0000 0110 0100 01012 = 06 4516

| 0000 | 0110 | 0100 | 0101 |
| --- | --- | --- | --- |

Аналогічно,щоб знайти -А:

1. Беру двійковий код додатнього А:

| 0000 | 0000 | 0001 | 0000 |
| --- | --- | --- | --- |

1. Інвертую кожен біт:

| 1111 | 1111 | 1110 | 1111 |
| --- | --- | --- | --- |

1. Додаю 1:

| 1111 | 1111 | 1111 | 0000 |
| --- | --- | --- | --- |

**-А** = -1610 = 1111 1111 1111 00002 = FF F016

Шукаю -В за такою ж логікою:

1. Беру двійковий код додатнього В:

| 0000 | 0110 | 0100 | 0101 |
| --- | --- | --- | --- |

1. Інвертую кожен біт:

| 1111 | 1001 | 1011 | 1010 |
| --- | --- | --- | --- |

1. Додадю 1:

| 1111 | 1001 | 1011 | 1011 |
| --- | --- | --- | --- |

**-В** = -160510 = 1111 1001 1011 10112 = F9 BB16

**4) числа A, B, C, -A , -B и -C у чотирьохбайтовому форматі Shortlnt;**

**А** = 1610 = 0000 0000 0000 0000 0000 0000 0001 00002 = 00 00 00 1016

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0001 | 0000 |
| --- | --- | --- | --- | --- | --- | --- | --- |

**В** = 160510 = 0000 0000 0000 0000 0000 0110 0100 01012 = 00 00 06 4516

| 0000 | 0000 | 0000 | 0000 | 0000 | 0110 | 0100 | 0101 |
| --- | --- | --- | --- | --- | --- | --- | --- |

**С** = 1605200510 = 0000 0000 1111 0100 1110 1111 0010 01012 = 00 F4 EF 2516

| 0000 | 0000 | 1111 | 0100 | 1110 | 1111 | 0010 | 0101 |
| --- | --- | --- | --- | --- | --- | --- | --- |

Шукаю -А:

1. Беру двійковий код додатнього А:

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0001 | 0000 |
| --- | --- | --- | --- | --- | --- | --- | --- |

1. Інвертую кожен біт:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1110 | 1111 |
| --- | --- | --- | --- | --- | --- | --- | --- |

1. Додаю 1:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 0000 |
| --- | --- | --- | --- | --- | --- | --- | --- |

**-А** = -1610 = 1111 1111 1111 1111 1111 1111 1111 00002 = FF FF FF F016

Рахую -В:

1. Беру двійковий код додатнього В:

| 0000 | 0000 | 0000 | 0000 | 0000 | 0110 | 0100 | 0101 |
| --- | --- | --- | --- | --- | --- | --- | --- |

1. Інвертую кожен біт:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1001 | 1011 | 1010 |
| --- | --- | --- | --- | --- | --- | --- | --- |

1. Додаю 1:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1001 | 1011 | 1011 |
| --- | --- | --- | --- | --- | --- | --- | --- |

**-В** = -160510 = 1111 1111 1111 1111 1111 1001 1011 10112 = FF FF F9 BB16

Таким же спосібом -С:

1. Беру двійковий код додатнього С:

| 0000 | 0000 | 1111 | 0100 | 1110 | 1111 | 0010 | 0101 |
| --- | --- | --- | --- | --- | --- | --- | --- |

1. Інвертую кожен біт:

| 1111 | 1111 | 0000 | 1011 | 0001 | 0000 | 1101 | 1010 |
| --- | --- | --- | --- | --- | --- | --- | --- |

1. Додаю 1:

| 1111 | 1111 | 0000 | 1011 | 0001 | 0000 | 1101 | 1011 |
| --- | --- | --- | --- | --- | --- | --- | --- |

**-С** = -1605200510 = 1111 1111 0000 1011 0001 0000 1101 10112 = FF 0B 10 DB16

**5) числа A, B, C, -A , -B и -C у восьмибайтовому форматі Longlnt ;**

**А** = 1610 = 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0001 00002 = 00 00 00 00 00 00 00 1016

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0001 | 0000 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**B** = 160510 = 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0110 0100 01012 = 00 00 00 00 00 00 06 4516

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0110 | 0100 | 0101 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**C** = 1605200510 = 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 1111 0100 1110 1111 0010 01012 = 00 00 00 00 00 F4 EF 2516

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 1111 | 0100 | 1110 | 1111 | 0010 | 0101 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

Для розрахунку -А:

1. Беру двійковий код додатнього А:

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0001 | 0000 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

1. Інвертую кожен біт:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1110 | 1111 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

1. Додаю 1:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 0000 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**-А** = -1610 = 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 00002 = FF FF FF FF FF FF FF F016

Також,для -В:

1. Беру двійковий код додатнього В:

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0110 | 0100 | 0101 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

1. Інвертую кожен біт:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1001 | 1011 | 1010 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

1. Додаю 1:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1001 | 1011 | 1011 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**-В** = -160510 = 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1001 1011 10112 = FF FF FF FF FF FF F9 BB16

І для -С:

1. Беру двійковий код додатнього С:

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 1111 | 0100 | 1110 | 1111 | 0010 | 0101 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

1. Інвертую кожен біт:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 0000 | 1011 | 0001 | 0000 | 1101 | 1010 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

1. Додаю 1:

| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 0000 | 1011 | 0001 | 0000 | 1101 | 1011 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

**-С** = -1605200510 = 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 0000 1011 0001 0000 1101 10112 = FF FF FF FF FF 0B 10 DB16

**4. Представити числа D, E и F у нормалізованому вигляді:**

**D** = 0,00210 = 0,00000000102

1. Переводжу число до наукового запису: 0,00000000102 = 1,0 \* (-9,адже я зсуваю кому вправо на 9 позицій, що відповідає множенню на 2 в ступені -9.)
2. Z(знак)=+0(так,як число додатнє), M(мантиса) = 1,0, N(порядок) = 2, q(експонента) = -9

**E** = 0,18310 = 0,00101110112

1. Переводжу число до наукового запису: 0,00101110112 = 1,0111011 \* (аналогічно -3 ступень через зсув коми вправо на 3 позиції)
2. Z(знак)=+0(так,як число додатнє), M(мантиса) = 1,0111011, N(порядок) = 2, q(експонента) = -3

**F** = 1832,42110 = 11100101000,01101011112

1. Також переводжу до нормального запису: 11100101000,01101011112 = 1,11001010000110101111 \* (у цьому випадку кому я зсувала вліво,тому 10)
2. Z(знак)=+0(так,як число додатнє), M(мантиса) = 1,11001010000110101111, N(порядок) = 2, q(експонента) = 10

**5. За допомогою розрядної сітки показати в звіті представлення дійсних чисел в наступних форматах:**

**1)Числа D и –D у чотирьохбайтовому форматі Single (float);**

При представленні чисел у форматі IEEE 754, біти розподіляються між знаком, порядком і мантисою відповідно до стандарту.

У форматі Single(float) 32 біти відводиться під число, яке складається зі знаку (1 біт), експоненти (8 біт) і мантиси (23 біти).Зсув = 127

D = 0,00210 = 0,00000000102

Отже:

1. Знак = 0(число додатнє)
2. Мантиса = 1,0 -> 0
3. Порядок = q + 127(зсув для представлення у чотирьохбайтовому форматі Single (float) IEEE 754) = -9 + 127 = 11810 = 11101102
4. Розрядна сітка:

| 31-й біт | 23-30 біти | 0-22 біти |
| --- | --- | --- |
| 0 | 01110110 | 00000000000000000000000 |
| Знак | Порядок | Мантиса |

Щоб отримати D у чотирьохбайтовому форматі Single (float) потрібно:

Знак + Порядок + Мантиса

Отже,

**D** = 0011 1011 0000 0000 0000 0000 0000 0000 = 3B 00 00 00

-D = - 0,00210 = 0,00000000102

Отже:

1. Знак = 1(так як число від’ємне)
2. Мантиса = 1,0 -> 0
3. Порядок = q + 127(зсув для представлення у чотирьохбайтовому форматі Single (float) IEEE 754) = -9 + 127 = 11810 = 11101102
4. Розрядна сітка:

| 31-й біт | 23-30 біти | 0-22 біти |
| --- | --- | --- |
| 1 | 01110110 | 00000000000000000000000 |
| Знак | Порядок | Мантиса |

Щоб отримати -D у чотирьохбайтовому форматі Single (float) потрібно:

Знак + Порядок + Мантиса

Отже,

**-D** = 1011 1011 0000 0000 0000 0000 0000 0000 = BB 00 00 00

**2)Числа E и –E у восьмибайтовому форматі Double (double);**

У форматі Double (double) 64 біти відводиться під число, яке складається зі знаку (1 біт), експоненти (11 біт) і мантиси (52 біти).Зсув = 1023

E= 0,18310 = 0,00101110112

Таким же чином:

1. Знак = 0(число додатнє)
2. Мантиса = 1,0111011 -> 0111011
3. Порядок = q + 1023(зсув для представлення у восьмибайтовому форматі Double (double) IEEE 754) = -3 + 1023 = 102010 = 11111111002
4. Розрядна сітка:

| 63-й біт | 52-62 біти | 0-51 біти |
| --- | --- | --- |
| 0 | 01111111100 | 0111011000000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

Щоб отримати E у восьмибайтовому форматі Double (double) потрібно:

Знак + Порядок + Мантиса

Отже,

**E** = 0011 1111 1100 0111 0110 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 = 3F C7 60 00 00 00 00 00

-E= -0,18310 = 0,00101110112

За аналогією:

1. Знак = 1(число від’ємне)
2. Мантиса = 1,0111011 -> 0111011
3. Порядок = q + 1023(зсув для представлення у восьмибайтовому форматі Double (double) IEEE 754) = -3 + 1023 = 102010 = 11111111002
4. Розрядна сітка:

| 63-й біт | 52-62 біти | 0-51 біти |
| --- | --- | --- |
| 1 | 01111111100 | 0111011000000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

Також,щоб отримати -E у восьмибайтовому форматі Double (double) потрібно:

Знак + Порядок + Мантиса

**-Е** = 1011 1111 1100 0111 0110 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 = BF C7 60 00 00 00 00 00

**3)Числа F и –F у десятибайтовому форматі Extended (long double).;**

У форматі Extended (long double)80 біт відводиться під число, яке складається зі знаку (1 біт), експоненти (15 біт) і мантиси (64 біти).Зсув = 16383

F = 1832,42110 = 11100101000,01101011112

Також як і раніше:

1. Знак = 0(число додатнє)
2. Мантиса = 1,11001010000110101111 -> 111001010000110101111
3. Порядок = q + 16383(зсув для представлення у десятибайтовому форматі Extended (long double) IEEE 754) = 10 + 16383 = 1639310 = 1000000000010012
4. Розрядна сітка:

| 79-й біт | 64-78 біти | 0-63 біти |
| --- | --- | --- |
| 0 | 100000000001001 | 1110010100001101011110000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

Знак + Порядок + Мантиса

**F** = 0100 0000 0000 1001 1110 0101 0000 1101 0111 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 = 40 09 E5 0D 78 00 00 00 00 00

-F = -1832,42110 = 11100101000,01101011112

За аналогією:

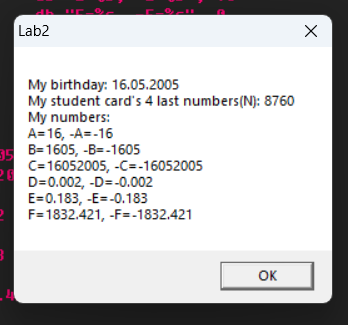
1. Знак = 1(число від’ємне)
2. 1,11001010000110101111 -> 111001010000110101111
3. Порядок = q + 16383(зсув для представлення у десятибайтовому форматі Extended (long double) IEEE 754) = 10 + 16383 = 1639310 = 1000000000010012
4. Розрядна сітка:

| 79-й біт | 64-78 біти | 0-63 біти |
| --- | --- | --- |
| 1 | 100000000001001 | 1110010100001101011110000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

Знак + Порядок + Мантиса

**-F** = 1100 0000 0000 1001 1110 0101 0000 1101 0111 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 = C0 09 E5 0D 78 00 00 00 00 00

**6.Скріншот виконання програми:**

****

**7.У режимі консолі виконати формування розширеного лістингу програми у вигляді файлу з розширенням “lst” за допомогою опції /Fl компілятора ML. Знайти в лістингу фрагмент, що містить коди команд і даних, і привести цей фрагмент в звіті з виконання лаб. роботи. У наведеному фрагменті лістингу знайти і помітити кольоровим олівцем або фломастером шістнадцяткові коди усіх варіантів представлення чисел A, B, … F.**

C

00000000 .data

00000000 00000100 [ buff db 256 dup(?)

00

]

00000100 00000020 [ buffD db 32 dup(?)

00

]

00000120 00000020 [ buffE db 32 dup(?)

00

]

00000140 00000020 [ buffF db 32 dup(?)

00

]

00000160 00000020 [ buffMinusD db 32 dup(?)

00

]

00000180 00000020 [ buffMinusE db 32 dup(?)

00

]

000001A0 00000020 [ buffMinusF db 32 dup(?)

00

]

000001C0 4C 61 62 32 00 title\_of\_messagebox db "Lab2", 0

000001C5 4D 79 20 62 69 data\_of\_messagebox db "My birthday: 16.05.2005", 10

72 74 68 64

61 79 3A 20

31 36 2E 30

35 2E 32 30

30 35 0A

000001DD 4D 79 20 73 74 db "My student card's 4 last numbers(N): 8760", 10

75 64 65 6E

74 20 63 61

72 64 27 73

20 34 20 6C

61 73 74 20

6E 75 6D 62

65 72 73 28

4E 29 3A 20

38 37 36 30

0A

00000207 4D 79 20 6E 75 db "My numbers:",10

6D 62 65 72

73 3A 0A

00000213 41 3D 25 64 2C db "A=%d, -A=%d", 10

20 2D 41 3D

25 64 0A

0000021F 42 3D 25 64 2C db "B=%d, -B=%d", 10

20 2D 42 3D

25 64 0A

0000022B 43 3D 25 64 2C db "C=%d, -C=%d", 10

20 2D 43 3D

25 64 0A

00000237 44 3D 25 73 2C db "D=%s, -D=%s", 10

20 2D 44 3D

25 73 0A

00000243 45 3D 25 73 2C db "E=%s, -E=%s", 10

20 2D 45 3D

25 73 0A

0000024F 46 3D 25 73 2C db "F=%s, -F=%s", 0

20 2D 46 3D

25 73 00

0000025B 10 A db 16

0000025C 0010 PlusA1 dw 16

0000025E 00000010 PlusA2 dd 16

00000262 PlusA3 dq 16

0000000000000010

0000026A F0 MinusA db -16

0000026B FFF0 MinusA1 dw -16

0000026D FFFFFFF0 MinusA2 dd -16

00000271 MinusA3 dq -16

FFFFFFFFFFFFFFF0

00000279 0645 B dw 1605

0000027B 00000645 PlusB1 dd 1605

0000027F PlusB2 dq 1605

0000000000000645

00000287 F9BB MinusB dw -1605

00000289 FFFFF9BB MinusB1 dd -1605

0000028D MinusB2 dq -1605

FFFFFFFFFFFFF9BB

00000295 00F4EF25 PlusC dd 16052005

00000299 PlusC1 dq 16052005

0000000000F4EF25

000002A1 FF0B10DB MinusC dd -16052005

000002A5 MinusC1 dq -16052005

FFFFFFFFFF0B10DB

000002AD D dq 0.002

3F60624DD2F1A9FC

000002B5 3B03126F PlusD1 dd 0.002

000002B9 MinusD dq -0.002

BF60624DD2F1A9FC

000002C1BB03126F MinusD1 dd 0.002

000002C5 E dq 0.183

3FC76C8B43958106

000002CD MinusE dq -0.183

BFC76C8B43958106

000002D5 F dq 1832.421

409CA1AF1A9FBE77

000002DD MinusF dq -1832.421

C09CA1AF1A9FBE77

000002E5 PlusF1 dt 1832.421

4009E50D78D4FDF3B646

000002EF MinusF1 dt -1832.421

C009E50D78D4FDF3B646

00000000 .code

00000000 Main:

invoke FloatToStr2, D, addr buffD

invoke FloatToStr2, MinusD, addr buffMinusD

invoke FloatToStr2, E, addr buffE

invoke FloatToStr2, F, addr buffF

invoke FloatToStr2, MinusE, addr buffMinusE

invoke FloatToStr2, MinusF, addr buffMinusF

invoke wsprintf, addr buff, addr data\_of\_messagebox, PlusA2, MinusA2, PlusB1, MinusB1, PlusC, MinusC, addr buffD, addr buffMinusD, addr buffE, addr buffMinusE, addr buffF, addr buffMinusF

invoke MessageBox, 0, addr buff, addr title\_of\_messagebox, 0

invoke ExitProcess, 0

end Main

**8.Порівняння всіх результатів:**

| Число(формат) | Значення | Мої розрахунки | Дані з лістингу програми |
| --- | --- | --- | --- |
| “ddmmyyyy” | 16.05.2005 | 1 -> 3116  6 -> 3616  0 -> 3016  5 -> 3516  2 -> 3216  0 -> 3016  0 -> 3016  5 -> 3516 | 31,36,30,35,32,30,30,35 |
| Цілі числа | | | |
| А(двобайтовий) | 16 | 00 1016 | 0010 |
| А(чотирьохбайтовий) | 16 | 00 00 00 1016 | 00000010 |
| А(восьмибайтовий) | 16 | 00 00 00 00 00 00 00 1016 | 0000000000000010 |
| -А(однобайтовий) | -16 | F016 | F0 |
| -А(двобайтовий) | -16 | FF F016 | FFF0 |
| -А(чотирьохбайтовий) | -16 | FF FF FF F016 | FFFFFFF0 |
| -А(восьмибайтовий) | -16 | FF FF FF FF FF FF FF F016 | FFFFFFFFFFFFFFF0 |
| B(двобайтовий) | 1605 | 06 4516 | 0645 |
| В(чотирьохбайтовий) | 1605 | 00 00 06 4516 | 00000645 |
| В(восьмибайтовий) | 1605 | 00 00 00 00 00 00 06 4516 | 0000000000000645 |
| -В(двобайтовий) | -1605 | F9 BB16 | F9BB |
| -В(чотирьохбайтовий) | -1605 | FF FF F9 BB16 | FFFFF9BB |
| -В(восьмибайтовий) | -1605 | FF FF FF FF FF FF F9 BB16 | FFFFFFFFFFFFF9BB |
| С(чотирьохбайтовий) | 16052005 | 00 F4 EF 2516 | 00F4EF25 |
| С(восьмибайтовий) | 16052005 | 00 00 00 00 00 F4 EF 2516 | 0000000000F4EF25 |
| -С(чотирьохбайтовий) | -16052005 | FF 0B 10 DB16 | FF0B10DB |
| -С(восьмибайтовий) | -16052005 | FF FF FF FF FF 0B 10 DB16 | FFFFFFFFFF0B10DB |
| Дійсні числа | | | |
| D(чотирьохбайтовий) | 0,002 | 3B 00 00 00 | 3B03126F |
| -D(чотирьохбайтовий) | -0,002 | BB 00 00 00 | BB03126F |
| E(восьмибайтовий) | 0,183 | 3F C2 EC 00 00 00 00 00  3F C7 60 00 00 00 00 00 | 3FC76C8B43958106 |
| -E(восьмибайтовий) | -0,183 | ВF C2 EC 00 00 00 00 00  BF C7 60 00 00 00 00 00 | BFC76C8B43958106 |
| F(десятибайтовий) | 1832,421 | 40 09 E5 0D 78 00 00 00 00 00 | 4009E50D78D4FDF3B646 |
| -F(десятибайтовий) | -1832,421 | C0 09 E5 0D 78 00 00 00 00 00 | C009E50D78D4FDF3B646 |

Отже,як можна побачити за результатами,цілі числа співпадають повністю,а дійсні через закруглення до 10 знаків після коми не до кінця.

**9.Висновок:**

Протягом виконання лабораторної роботи я поліпшила свої знання форматів внутрішнього представлення цілих і дійсних чисел, попрактикувалась у представленні чисел у однобайтовому, двобайтовому, чотирьохбайтовому, восьмибайтовому та десятибайтовому форматі, навчилась представляти числа у нормалізованому вигляді. Більше дізналась про формат IEEE 754 . Порівнюючи результати, отримані моїми розрахунками з результатами, що містяться в лістингу програми,на щастя,можу сказати що вони збігаються.Також під час розробки простої програми на Masm32 з вивідом цілих та дійсних чисел у MessageBox я попрацювала з новими для себе функціями в masm32,а саме FloatToStr,FloatToStr2 та wsprintf.В цілому,можу сказати,що тепер я розумію більш глибоко,як компьютер працює з цілими та дійсними числами.