**Міністерство освіти і науки України**

**Національний технічний університет України**

**«Київський політехнічний інститут ім. 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»**

**Виконав:** **Перевірив:**

Студент 2 курсу ФІОТ-у доц. Павлов В.Г.

групи ІМ-21

Лесько Дмитро Миколайович

номер у списку групи - 14

**Київ 2024**

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

Виконання завдань:

1. **Сформовані початкові числа:**

**Моя дата народження**: 08.02.2005

**Номер моєї залікової книжки студента**: 9076 (N)

Початкові числа, сформовані на основі початкових даних, введених вище:

* А- 08
* В- 0802
* С- 08022005

Далі робимо ділення для підрахування наступних початкових чисел (при діленні результат округлюємо до третього знаку після коми):

* D- 08 / 9076 = 0.00088145 ~ **0.001**
* E- 0802 / 9076 = 0.08836492 ~ **0.088**
* F- 08022005 / 9076 = 883.870097 ~ **833.87**

2. **Представляємо отримані початкові числа в двійковій системі числення (числа з дробовою частиною округлюємо до 10 знаків після коми):**

* А10 = 08, A2 = 1000
* B10 = 0802, B2 = 1100100010
* C10 = 08022005, C2 = 11110100110011111110101

Далі відповідно до вимог я буду демонструвати перетворення чисел з плаваючою комою з десяткової у двійкову систему:

* D10 = 0.001

Розрахунок:  
010 = 0/2 = 02- перевели цілу частину

Далі рахуємо дробову частину:

|  |  |  |
| --- | --- | --- |
| *Операція* | *результат* | ***число*** |
| ,001 \* 2 | ,002 | **0** |
| ,002 \* 2 | ,004 | **0** |
| ,004 \* 2 | ,008 | **0** |
| ,008 \* 2 | ,016 | **0** |
| ,016 \* 2 | ,032 | **0** |
| ,032 \* 2 | ,064 | **0** |
| ,064 \* 2 | ,128 | **0** |
| ,128 \* 2 | ,256 | **0** |
| ,256 \* 2 | ,512 | **0** |
| ,512 \* 2 | 1,*024* | **1** |
| ,024 \* 2 | ,048 | **0** |

,00110 = ,00000000012

0,00110 = 0,00000000012

* E10 = 0.088

Розрахунок:  
010 = 0/2 = 02- перевели цілу частину

Далі рахуємо дробову частину:

|  |  |  |
| --- | --- | --- |
| *Операція* | *результат* | ***число*** |
| ,088 \* 2 | ,176 | **0** |
| ,176 \* 2 | ,352 | **0** |
| ,352 \* 2 | ,704 | **0** |
| ,704 \* 2 | 1,408 | **1** |
| ,408 \* 2 | ,816 | **0** |
| ,816 \* 2 | 1,632 | **1** |
| ,632 \* 2 | 1,264 | **1** |
| ,264 \* 2 | ,528 | **0** |
| ,528 \* 2 | 1,056 | **1** |
| ,056 \* 2 | ,112 | **0** |
| ,112 \* 2 | ,224 | **0** |

,08810 = ,000101101002

0,08810 = 0,00010110102

* F10 = 833.87

Розрахунок:

Тепер для переводу цілої частини теж застосуємо таблицю:

|  |  |  |
| --- | --- | --- |
| *Операція* | *Результат (цілу число)* | ***Число (залишок)*** |
| 833 / 2 | 416 | **1** |
| 416 / 2 | 208 | **0** |
| 208 / 2 | 104 | **0** |
| 104 / 2 | 52 | **0** |
| 52 / 2 | 26 | **0** |
| 26 / 2 | 13 | **0** |
| 13 / 2 | 6 | **1** |
| 6 / 2 | 3 | **0** |
| 3 / 2 | 1 | **1** |
| 1 / 2 | 0 | **1** |

83310 = 11010000012

Далі рахуємо дробову частину:

|  |  |  |
| --- | --- | --- |
| *Операція* | *результат* | ***число*** |
| ,87 \* 2 | 1,74 | **1** |
| ,74 \* 2 | 1,48 | **1** |
| ,48 \* 2 | ,96 | **0** |
| ,96 \* 2 | 1,92 | **1** |
| ,92 \* 2 | 1,84 | **1** |
| ,84 \* 2 | 1,68 | **1** |
| ,68 \* 2 | 1,36 | **1** |
| ,36 \* 2 | ,72 | **0** |
| ,72 \* 2 | 1,44 | **1** |
| ,44\* 2 | ,88 | **0** |
| ,88 \* 2 | 1,76 | **1** |

,8710 = ,11011110102

833,8710 = 1101000001,11011110112

3. **Представлення цілих чисел і відповідних форматах за допомогою розрядної сітки:**

* 08.02.2005 у вигляді символьного рядка:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Символ** | 0 | 8 | 0 | 2 | 2 | 0 | 0 | 5 |
| **Hex** | 3016 | 3816 | 3016 | 3216 | 3216 | 3016 | 3016 | 3516 |

* Числа А і -А у форматі Byte:

А:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Рахуємо: | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

810 = 000010002 = 0816

-А:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Інвертуємо: | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| Додаємо одиницю: | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |

-810 = 111110002 = F816

* Числа у двобайтовому форматі Word:

A:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 |  | 0 | 0 | 0 | 0 |  | 0 | 0 | 0 | 0 |  | 1 | 0 | 0 | 0 |

810 = 00000000000010002 = 00 0816

B:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 |  | 0 | 0 | 1 | 1 |  | 0 | 0 | 1 | 0 |  | 0 | 0 | 1 | 0 |

80210 = 0000001100100010­2 = 03 2216

-A:

Інвертуємо:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 |  | 1 | 1 | 1 | 1 |  | 1 | 1 | 1 | 1 |  | 0 | 1 | 1 | 1 |

Додаємо одиницю:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 |  | 1 | 1 | 1 | 1 |  | 1 | 1 | 1 | 1 |  | 1 | 0 | 0 | 0 |

-810 = 11111111111110002 = FF F816

-B:

Інвертуємо:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 |  | 1 | 1 | 0 | 0 |  | 1 | 1 | 0 | 1 |  | 1 | 1 | 0 | 1 |

Додаємо одиницю:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 |  | 1 | 1 | 0 | 0 |  | 1 | 1 | 0 | 1 |  | 1 | 1 | 1 | 0 |

-80210 = 11111100110111102 = FC DE16

* Числа у чотирьохбайтовому форматі ShortInt:

A:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |

810 = 00000000 00000000 00000000 00000000 000010112 = 0000000B16

B:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |

80210 = 00000000 00000000 00000011 001000102 = 0000032216

C:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 |

802200510 = 00000000 01111010 01100111 111101012 = 007A67F516

-A:

Інвертуємо:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |

Додаємо 1:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |

-810 = 11111111 11111111 11111111 111110002 = FFFFFFF816

-B:

Інвертуємо:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 |

Додаємо 1:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |

-80210 = 11111111 11111111 11111100 110111102 = FFFFFCDE16

-C:

Інвертуємо:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

Додаємо одиницю

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |

-802200510 = 11111111 10000101 10011000 000010112 = FF85980B16

* Числа у восьмибайтовому форматі LongInt:

А:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 00000000 | 00000000 | 000000000 | 00000000 | 000000000 | 00000000 | 00000000 | 00001000 |

810 = 00000000 00000000 00000000 00000000 00000000 00000000 00000000 000010002 = 000000000000000816

В:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 00000000 | 00000000 | 00000000 | 00000000 | 00000000 | 00000000 | 00000011 | 00100010 |

80210 = 00000000 000000000 00000000 00000000 00000000 00000000 00000011 001000102 = 000000000000032216

С:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 00000000 | 00000000 | 00000000 | 00000000 | 00000000 | 01111010 | 01100111 | 11110101 |

802200510 = 00000000 00000000 00000000 00000000 00000000 01111010 01100111 111101012 = 00000000007A67F516

-A:

Інвертуємо:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11110111 |

Додаємо 1:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11111000 |

-810 = 11111111 11111111 11111111 11111111 11111111 11111111 11111111 111110002 = FFFFFFFFFFFFFFF816

-B:

Інвертуємо:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11111100 | 11011101 |

Додаємо 1:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 11111100 | 11011110 |

-80210 = 11111111 11111111 11111111 11111111 11111111 11111111 11111100 110111102 = FFFFFFFFFFFFFCDE16

-C:

Інвертуємо:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 10000101 | 10011000 | 00001010 |

Додаємо 1:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 11111111 | 11111111 | 11111111 | 11111111 | 11111111 | 10000101 | 10011000 | 00001011 |

-802200510 = 11111111 11111111 11111111 11111111 11111111 10000101 10011000 000010112 = FFFFFFFFFF85980B16

4. **Представлення чисел у нормалізованому вигляді:**

* D:

1. D = 0.00110 = 0.00000000012
2. 0. 00000000012 = 1.000 \* 2-10
3. ZN = +0, M = 1.000, N = 2, q = -10

* E:

1. E = 0.08810 = 0.00010110102
2. 0.00010110102 = 1.011010 \* 2-4
3. ZN = +0, M = 1.011010, N = 2, q = -4

* F:

1. F = 833.8710 = 1101000001.11011110112
2. 1101000001.11011110112 = 1.1010000011101111011 \* 29
3. ZN = +0, M = 1.1010000011101111011, N = 2, q = 9

5. **Представлення дійсних чисел у наступних форматах:**

* Числа D і -D у чотирьохбайтовому форматі Single float:

D = 0.00110

1. Знак числа- 0(додатній)
2. Порядок: q + 127 = -10 + 127 = 11710 = 11101012
3. Мантиса: 1.000 => 011010

Представлення у розрядній сітці:

|  |  |  |
| --- | --- | --- |
| 31 | 30-23 | 22-0 |
| 0 | 01110101 | 00000000000000000000000 |
| Знак | Порядок | Мантиса |

0011 1010 1000 0000 0000 0000 0000 0000 = **3A80000016**

-D = -0.00110

1. Знак числа- 1(від'ємний)
2. Порядок: q + 127 = -10 + 127 = 11710 = 11101012
3. Мантиса: 1.000 => 00000

Представлення у розрядній сітці:

|  |  |  |
| --- | --- | --- |
| 31 | 30-23 | 22-0 |
| 1 | 01110101 | 0000000000000000000000 |
| Знак | Порядок | Мантиса |

1011 1010 1000 0000 0000 0000 0000 0000 = **BA80000016**

* Числа Е і -Е у восьмибайтовому форматі Double:

E = 0.08810

1. Знак числа- 0(додатній)
2. Порядок: q + 1023 = -4 + 1023 = 101910 = 11111110112
3. Мантиса: 1.011010 => 011010

Представлення у розрядній сітці:

|  |  |  |
| --- | --- | --- |
| 63 | 62-52 | 51-0 |
| 0 | 01111111011 | 011010000000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

0011111110110110100000000000000000000000000000000000000000000000 = **3FB680000000000016**

E = -0.08810

1. Знак числа- 1(додатній)
2. Порядок: q + 1023 = -4 + 1023 = 101910 = 11111110112
3. Мантиса: 1.011010 => 011010

Представлення у розрядній сітці:

|  |  |  |
| --- | --- | --- |
| 63 | 62-52 | 51-0 |
| 1 | 01111111011 | 011010000000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

1011111110110110100000000000000000000000000000000000000000000000 = **BFB680000000000016**

* Числа F і -F у десятибайтовому форматі Extended:

F = 833.8710

1. Знак числа- 0(додатній)
2. Порядок: q + 16383 = 9 + 16383 = 1639210 = 1000000000010002
3. Мантиса: 1.1010000011101111011 => 11010000011101111011

Представлення у розрядній сітці:

|  |  |  |
| --- | --- | --- |
| 79 | 78-64 | 63-0 |
| 0 | 100000000001000 | 1101000001110111101100000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

01000000000010001101000001110111101100000000000000000000000000000000000000000000 = **4008D077B0000000000016**

F = -833.8710

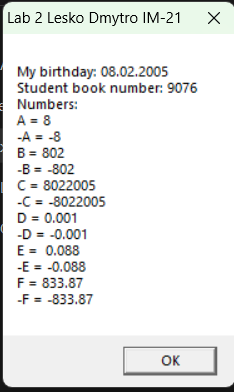
1. Знак числа- 1(додатній)
2. Порядок: q + 16383 = 9 + 16383 = 1639210 = 1000000000010002
3. Мантиса: 1.1010000011101111011 => 11010000011101111011

Представлення у розрядній сітці:

|  |  |  |
| --- | --- | --- |
| 79 | 78-64 | 63-0 |
| 1 | 100000000001000 | 1101000001110111101100000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

11000000000010001101000001110111101100000000000000000000000000000000000000000000 = **C008D077B0000000000016**

6. **На мові програмування Assembler у середовищі MASM32 я написав програму, яка виводить у віконному форматі усі варіанти констант, які були отримані на початку і протягом виконання цієї лабораторної роботи. Виконав компіляцію і у вікні MessageBox отримав перелік усіх потрібних констант**:



7. **За допомогою опцій компілятора ML /Fl провів розширений лістенінг і отримав файл з розширенням lst. Далі, відкривши файл, знайшов фрагмент, що містить коди команд і даних:**

00000000 .data?

00000000 00000100 [ textBoxBuffer db 256 dup(?)

00

]

00000100 00000080 [ positiveBufferD db 128 dup(?)

00

]

00000180 00000080 [ positiveBufferE db 128 dup(?)

00

]

00000200 00000080 [ positiveBufferF db 128 dup(?)

00

]

00000280 00000080 [ negativeBufferD db 128 dup(?)

00

]

00000300 00000080 [ negativeBufferE db 128 dup(?)

00

]

00000380 00000080 [ negativeBufferF db 128 dup(?)

00

]

00000000 .data

00000000 4C 61 62 20 32 messageBoxTitle db "Lab 2 Lesko Dmytro IM-21", 0

20 4C 65 73

6B 6F 20 44

6D 79 74 72

6F 20 49 4D

2D 32 31 00

00000019 4D 79 20 62 69 messageBoxContent db "My birthday: %s", 0Ah, "Student book number: %s", 0Ah, "Numbers:",0Ah, "A = %d", 0Ah, "-A = %d", 0Ah, "B = %d", 0Ah, "-B = %d", 0Ah,

72 74 68 64

61 79 3A 20

25 73 0A 53

74 75 64 65

6E 74 20 62

6F 6F 6B 20

6E 75 6D 62

65 72 3A 20

25 73 0A 4E

75 6D 62 65

72 73 3A 0A

41 20 3D 20

25 64 0A 2D

41 20 3D 20

25 64 0A 42

20 3D 20 25

64 0A 2D 42

20 3D 20 25

64 0A 43 20

3D 20 25 64

0A 2D 43 20

3D 20 25 64

0A 44 20 3D

20 25 73 0A

2D 44 20 3D

20 25 73 0A

45 20 3D 20

25 73 0A 2D

45 20 3D 20

25 73 0A 46

20 3D 20 25

73 0A 2D 46

20 3D 20 25

73 00

"C = %d", 0Ah, "-C = %d", 0Ah, "D = %s", 0Ah, "-D = %s", 0Ah, "E = %s", 0Ah, "-E = %s", 0Ah, "F = %s", 0Ah, "-F = %s", 0

000000A4 30 38 2E 30 32 myBirthday db "08.02.2005", 0

2E 32 30 30

35 00

000000AF 39 30 37 36 00 studentBookNumber db "9076", 0

000000B4 08 positiveByteA db 8

000000B5 0008 positiveWordA dw 8

000000B7 00000008 positiveShortIntA dd 8

000000BB positiveLongIntA dq 8

0000000000000008

000000C3 F8 negativeByteA db -8

000000C4 FFF8 negativeWordA dw -8

000000C6 FFFFFFF8 negativeShortIntA dd -8

000000CA negativeLongIntA dq -8

FFFFFFFFFFFFFFF8

000000D2 0322 positiveWordB dw 802

000000D4 00000322 positiveShortIntB dd 802

000000D8 positiveLongIntB dq 802

0000000000000322

000000E0 FCDE negativeWordB dw -802

000000E2 FFFFFCDE negativeShortIntB dd -802

000000E6 negativeLongIntB dq -802

FFFFFFFFFFFFFCDE

000000EE 007A67F5 positiveShortIntC dd 8022005

000000F2 positiveLongIntC dq 8022005

00000000007A67F5

000000FA FF85980B negativeShortIntC dd -8022005

000000FE negativeLongIntC dq -8022005

FFFFFFFFFF85980B

00000106 3A83126F positiveSingleD dd 0.001

0000010A BA83126F negativeSingleD dd -0.001

0000010E positiveDoubleD dq 0.001

3F50624DD2F1A9FC

00000116 negativeDoubleD dq -0.001

BF50624DD2F1A9FC

0000011E positiveDoubleE dq 0.088

3FB6872B020C49BA

00000126 negativeDoubleE dq -0.088

BFB6872B020C49BA

0000012E positiveDoubleF dq 833.87

408A0EF5C28F5C29

00000136 negativeDoubleF dq -833.87

C08A0EF5C28F5C29

0000013E positiveExtendedF dt 833.87

4008D077AE147AE147AE

00000148 negativeExtendedF dt -833.87

C008D077AE147AE147AE

00000000 .code

00000000 lab2Main:

00000000 DD 05 0000011E R fld positiveDoubleE

invoke FpuFLtoA, 0, 3, addr positiveBufferE, SRC1\_FPU or SRC2\_DIMM

00000019 DD 05 00000116 R fld negativeDoubleD

invoke FpuFLtoA, 0, 3, addr negativeBufferE, SRC1\_FPU or SRC2\_DIMM

invoke FloatToStr2, positiveDoubleD, addr positiveBufferD

invoke FloatToStr2, positiveDoubleF, addr positiveBufferF

invoke FloatToStr2, negativeDoubleD, addr negativeBufferD

invoke FloatToStr2, negativeDoubleF, addr negativeBufferF

invoke wsprintf, addr textBoxBuffer, addr messageBoxContent, addr myBirthday, addr studentBookNumber,

positiveShortIntA, negativeShortIntA,

positiveShortIntB, negativeShortIntB,

positiveShortIntC, negativeShortIntC,

addr positiveBufferD, addr negativeBufferD,

addr positiveBufferE, addr negativeBufferE,

addr positiveBufferF, addr negativeBufferF

invoke MessageBox, 0, addr textBoxBuffer, addr messageBoxTitle, 0

invoke ExitProcess, 0

end lab2Main

8. **Порівняння результатів з лістенінгу і тих, що ми отримали при обчислені у пунктах 3-6:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Назва константи | Формат | З розрахунків | З лістинінгу | Чи співпадають |
| ddmmyyyy | Символьний рядок | “0”-30  “8”-38  “0”-30  “2”-32  “2”-32  “0”-30  “0”-30  “5”-36 | 30 38 2E 30 32 2E 32 30 30 35 | Співпадають. У розрахунках я вказував символьний формат саме чисел. 2Е- це символьний формат крапки, яка з’явилася потім у програмі для зручнішого розділення дати народження. |
| А | Byte | 0816 | 08 | Так |
| -А | Byte | F816 | F8 | Так |
| А | Word | 00 0816 | 0008 | Так |
| -А | Word | FF F816 | FFF8 | Так |
| B | Word | 03 2216 | 0322 | Так |
| -В | Word | FC DE16 | FCDE | Так |
| А | ShortInt | 0000000816 | 00000008 | Так |
| -А | ShortInt | FFFFFFF816 | FFFFFFF8 | Так |
| В | ShortInt | 0000032216 | 00000322 | Так |
| -В | ShortInt | FFFFFCDE16 | FFFFFCDE | Так |
| С | ShortInt | 007A67F516 | 007A67F5 | Так |
| -С | ShortInt | FF85980B16 | FF85980B | Так |
| А | LongInt | 000000000000000816 | 0000000000000008 | Так |
| -А | LongInt | FFFFFFFFFFFFFFF816 | FFFFFFFFFFFFFFF8 | Так |
| В | LongInt | 000000000000032216 | 0000000000000322 | Так |
| -В | LongInt | FFFFFFFFFFFFFCDE16 | FFFFFFFFFFFFFCDE | Так |
| С | LongInt | 00000000007A67F516 | 00000000007A67F5 | Так |
| -С | LongInt | FFFFFFFFFF85980B16 | FFFFFFFFFF85980B | Так |
| D | Single | 3A80000016 | 3A83126F | Співпадає тільки перша частина (перші три цифри) через те, що дробову частину округлили до 10 знаків після коми |
| -D | Single | BA80000016 | BA83126F | Співпадає тільки перша частина (перші три цифри) через те, що дробову частину округлили до 10 знаків після коми |
| Е | Double | 3FB680000000000016 | 3FB6872B020C49BA | Співпадають перші 5 цифр, через округлення до 10 цифр після коми |
| -Е | Double | BFB680000000000016 | BFB6872B020C49BA | Співпадають перші 5 цифр, через округлення до 10 цифр після коми |
| F | Extended | 4008D077B0000000000016 | 4008D077AE147AE147AE | Співпадають перші вісім цифр, через округлення до 10 цифр після коми |
| -F | Extended | C008D077B0000000000016 | C008D077AE147AE147AE | Співпадають перші вісім цифр, через округлення до 10 цифр після коми |

9. **Висновок**:

Проробивши роботу з ручного переведення чисел у двійковий та шістнадцятковий формати різної довжини, я детально дослідив те, як обчислювальні машини працюють з числами. Побачив, що чим більше розрядність числа- тим більша його точність, хоч і розмір використовуваної пам’яті також буде більшим. Також дізнався нові дані про символьний формат у Hex.

Розробив нову програму, отримав новий та корисний досвід роботи з мовою Assembler, попрацював з баферами та новими можливостями платформи.

Провівши лістенінг та порівняння отриманих результатів з результатами файлу lst я ще більше зрозумів етапи перетворення програм у машинний код. Також зрозумів причини розбіжності результатів ручного обчислення і машинної обробки через округлення чисел після коми.