**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”

Виконав: Перевірив:

Студент ІІ курсу ФІОТ доц. Павлов В. Г.

групи ІМ-22

Сачко Максим Євгенійович

номер у списку групи (варіант): 20

Київ 2024

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

**1. Сформувати наступні початкові числа на основі дати народження студента, записаної у форматі dd.mm.yyyy:**

*Дата мого народження: 07.04.2005. Номер залікової книжки: 9148.*

* A – ціле двозначне число, що дорівнює dd;

A = 07

* B – ціле чотиризначне число, яке дорівнює ddmm, записаним без роздільника;

B = 0704

* C – ціле восьмизначне число, яке дорівнює ddmmyyyy, записаним без роздільника;

C = 07042005

Подальші числа D, E, F отримано шляхом ділення чисел A, B і C відповідно на номер залікової книжки, тобто є дійсними числами, які за завданням мають бути округлені до 3 знаків після коми:

* D – число, отримане шляхом ділення числа A на число N, де N тут і надалі – чотиризначний номер залікової книжки;

D = 07 / 9148 = 0,000765194578049847 ≈ 0,001

* E – число, отримане шляхом ділення числа B на число N;

E = 0704 / 9148 = 0,0769567118495846 ≈ 0,077

* F – число, отримане шляхом ділення числа C на число N;

F = 07042005 / 9148 = 769,786292085702 ≈ 769,786

**2. Представити отримані числа в двійковій системі числення (дробову частину округлювати до 10 знаків після коми).**

* A = 0710 = 01112
* B = 070410 = 10110000002
* C = 0704200510 = 110101101110011110101012
* D = 0,00110

010 = 02

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

Так як одинадцятий знак після коми дорівнює 0, інші 10 знаків залишаємо, як є.

,00110 = ,00000000012

D = 0,00110 = 0,00000000012

* E = 0,07710

010 = 02

|  |  |  |
| --- | --- | --- |
| ,077 \* 2 = | 0 | ,154 |
| ,154 \* 2 = | 0 | ,308 |
| ,308 \* 2 = | 0 | ,616 |
| ,616 \* 2 = | 1 | ,232 |
| ,232 \* 2 = | 0 | ,464 |
| ,464 \* 2 = | 0 | ,928 |
| ,928 \* 2 = | 1 | ,856 |
| ,856 \* 2 = | 1 | ,712 |
| ,712 \* 2 = | 1 | ,424 |
| ,424 \* 2 = | 0 | ,848 |
| ,848 \* 2 = | 1 | ,696 |

Так як одинадцятий знак після коми дорівнює 1, округлюємо.

,07710 = ,00010011112

E = 0,07710 = 0,00010011112

* F = 769,78610

76910 = 11000000012

|  |  |  |
| --- | --- | --- |
| ,786 \* 2 = | 1 | ,572 |
| ,572 \* 2 = | 1 | ,144 |
| ,144 \* 2 = | 0 | ,288 |
| ,288 \* 2 = | 0 | ,576 |
| ,576 \* 2 = | 1 | ,152 |
| ,152 \* 2 = | 0 | ,304 |
| ,304 \* 2 = | 0 | ,608 |
| ,608 \* 2 = | 1 | ,216 |
| ,216 \* 2 = | 0 | ,432 |
| ,432 \* 2 = | 0 | ,864 |
| ,864 \* 2 = | 1 | ,728 |

Так як одинадцятий знак після коми дорівнює 1, округлюємо.

,78610 = ,11001001012

F = 769,78610 = 1100000001,11001001012

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

* “ddmmyyyy” у вигляді символьного рядка:

“ddmmyyyy” = 07042005

‘0’ = 3016

‘7’ = 3716

‘0’ = 3016

‘4’ = 3416

‘2’ = 3216

‘0’ = 3016

‘0’ = 3016

‘5’ = 3516

* числа A и –A у однобайтовому форматі Byte:
  + A:

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

A = 0710 = 0000 01112 = 0716

* + -A:

1. інвертуємо A:

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

1. додаємо до інвертованого A одиницю:

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

-A = -0710 = 1111 10012 = F916

* числа A, B, -A и -B у двобайтовому формат Word:
  + A:

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

A = 0710 = 0000 0000 0000 01112 = 000716

* + B:

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

B = 070410 = 0000 0010 1100 00002 = 02C016

* + -A:

1. інвертуємо A:

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

1. додаємо до інвертованого A одиницю:

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

-A = -0710 = 1111 1111 1111 10012 = FFF916

* + -B:

1. інвертуємо B:

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

1. додаємо до інвертованого B одиницю:

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

-B = -070410 = 1111 1101 0100 00002 = FD4016

* числа A, B, C, -A , -B и -C у чотирьохбайтовому форматі Shortlnt:
  + 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 | 0 | 1 | 1 | 1 |

A = 0710 = 0000 0000 0000 0000 0000 0000 0000 01112 = 00 00 00 0716

* + B:

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

B = 070410 = 0000 0000 0000 0000 0000 0010 1100 00002 = 00 00 02 C016

* + C:

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

C = 0704200510 = 0000 0000 0110 1011 0111 0011 1101 01012 = 00 6B 73 D516

* + -A:

1. інвертуємо A:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 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 |

1. додаємо до інвертованого A одиницю:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 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 | 1 |

-A = -0710 = 1111 1111 1111 1111 1111 1111 1111 10012 = FF FF FF F916

* + -B:

1. інвертуємо B:

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

1. додаємо до інвертованого B одиницю:

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

-B = -070410 = 1111 1111 1111 1111 1111 1101 0100 00002 = FF FF FD 4016

* + -C:

1. інвертуємо C:

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

1. додаємо до інвертованого C одиницю:

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

-C = -0704200510 = 1111 1111 1001 0100 1000 1100 0010 10112 = FF 94 8C 2B16

* числа A, B, C, -A , -B и -C у восьмибайтовому форматі Longlnt:
  + A:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0111 |

A = 0710 = 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 01112 = 00 00 00 00 00 00 00 0716

* + B:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0010 | 1100 | 0000 |

B = 070410 = 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0010 1100 00002 = 00 00 00 00 00 00 02 C016

* + C:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0110 | 1011 | 0111 | 0011 | 1101 | 0101 |

C = 0704200510 = 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0110 1011 0111 0011 1101 01012 = 00 00 00 00 00 6B 73 D516

* + -A:

1. інвертуємо A:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1000 |

1. додаємо до інвертованого A одиницю:

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

-A = -0710 = 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 10012 = FF FF FF FF FF FF FF F916

* + -B:

1. інвертуємо B:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1101 | 0011 | 1111 |

1. додаємо до інвертованого B одиницю:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1101 | 0100 | 0000 |

-B = -070410 = 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1101 0100 00002 = FF FF FF FF FF FF FD 4016

* + -C:

1. інвертуємо C:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1001 | 0100 | 1000 | 1100 | 0010 | 1010 |

1. додаємо до інвертованого C одиницю:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1001 | 0100 | 1000 | 1100 | 0010 | 1011 |

-C = -0704200510 = 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1001 0100 1000 1100 0010 10112 = FF FF FF FF FF 94 8C 2B16

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

* D

1. D = 0,00110 = 0,00000000012
2. 0,00000000012 = 1 \* 2-10
3. ZN=+0, M = 1, N = 2, q = -10

* E

1. E = 0,07710 = 0,00010011112
2. 0,00010011112 = 1,001111 \* 2-4
3. ZN=+0, M = 1,001111, N = 2, q = -4

* F

1. F = 769,78610 = 1100000001,11001001012
2. 1100000001,11001001012 = 1,1000000011100100101 \* 29
3. ZN=+0, M = 1,1000000011100100101, N = 2, q = 9

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

* числа D и –D в у чотирьохбайтовому форматі Single (float);
  + D = 0,00110

1. знак числа – 0 (додатній)
2. порядок: q + 127 = -10 + 127 = 11710 = 11101012
3. мантиса: 1,0 → 0
4. розрядна сітка:

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

1. отже:

0011 1010 1000 0000 0000 0000 0000 00002 = 3A 80 00 0016

* + -D = -0,00110

1. знак числа – 1 (від’ємний)
2. порядок: q + 127 = -10 + 127 = 11710 = 11101012
3. мантиса: 1,0 → 0
4. розрядна сітка:

|  |  |  |
| --- | --- | --- |
| 31 | 30 23 | 22 0 |
| 1 | 01110101 | 00000000000000000000000 |
| Знак | Порядок | Мантиса |

1. отже:

1011 1010 1000 0000 0000 0000 0000 00002 = BA 80 00 0016

* числа E и –E у восьмибайтовому форматі Double (double);
  + E = 0,07710

1. знак числа – 0 (додатній)
2. порядок: q + 1023 = -4 + 1023 = 101910 = 11111110112
3. мантиса: 1,001111 → 001111
4. розрядна сітка:

|  |  |  |
| --- | --- | --- |
| 63 | 62 52 | 51 0 |
| 0 | 01111111011 | 0011110000000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

1. отже:

0011 1111 1011 0011 1100 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 00002 = 3F B3 C0 00 00 00 00 0016

* + -E = -0,07710

1. знак числа – 1 (від’ємний)
2. порядок: q + 1023 = -4 + 1023 = 101910 = 11111110112
3. мантиса: 1,001111 → 001111
4. розрядна сітка:

|  |  |  |
| --- | --- | --- |
| 63 | 62 52 | 51 0 |
| 1 | 01111111011 | 0011110000000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

1. отже:

1011 1111 1011 0011 1100 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 00002 = BF B3 C0 00 00 00 00 0016

* числа F и –F у десятибайтовому форматі Extended (long double).;
  + F = 769,78610

1. знак числа – 0 (додатній)
2. порядок: q + 16383 = 9 + 16383 = 1639210 = 1000000000010002
3. мантиса: 1,1000000011100100101 → 11000000011100100101
4. розрядна сітка:

|  |  |  |
| --- | --- | --- |
| 79 | 78 64 | 63 0 |
| 0 | 100000000001000 | 1100000001110010010100000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

1. отже:

0100 0000 0000 1000 1100 0000 0111 0010 0101 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 00002 = 40 08 C0 72 50 00 00 00 00 0016

* + -F = -769,78610

1. знак числа – 1 (від’ємний)
2. порядок: q + 16383 = 9 + 16383 = 1639210 = 1000000000010002
3. мантиса: 1,1000000011100100101 → 11000000011100100101
4. розрядна сітка:

|  |  |  |
| --- | --- | --- |
| 79 | 78 64 | 63 0 |
| 1 | 100000000001000 | 1100000001110010010100000000000000000000000000000000000000000000 |
| Знак | Порядок | Мантиса |

1. отже:

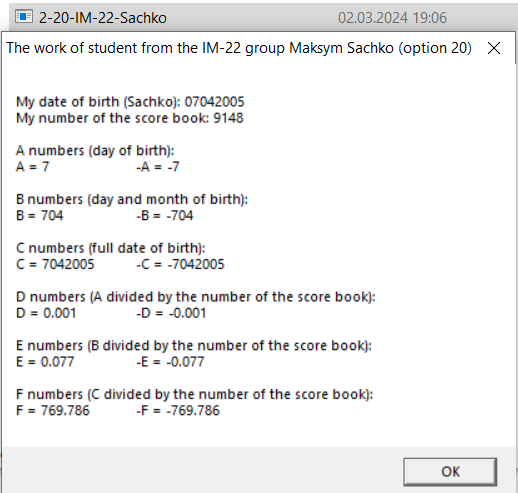
1100 0000 0000 1000 1100 0000 0111 0010 0101 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 00002 = C0 08 C0 72 50 00 00 00 00 0016

**6. Остаточні результати по кожному пункту надати у звіті окрім двійкової ще і в шістнадцятковій системі числення.**

Завдання виконане в попередніх розділах.

**7. Скласти програму на мові Assembler, в якій задати вказані символи і усі варіанти констант A, B, ... F, які присутні у пунктах 3 та 5. Значення констант A, B, ... F вивести у віконній формі на екран.**

**8. У режимі консолі виконати компіляцію і виконання програми. Привести скриншот вікна виведення значень змінних в звіті.**



**9. У режимі консолі виконати формування розширеного лістингу програми у вигляді файлу з розширенням “lst” за допомогою опції /Fl компілятора ML.**

**10. Знайти в лістингу фрагмент, що містить коди команд і даних, і привести цей фрагмент в звіті з виконання лаб. роботи.**

Фрагмент лістингу буде наведений у завданні нижче.

**11. У наведеному фрагменті лістингу знайти і помітити кольоровим олівцем або фломастером шістнадцяткові коди усіх варіантів представлення чисел A, B, … F.**

00000000 .data?

00000000 00000400 [ SachkoOlimpusFinalBufferForAll db 1024 dup(?)

00

]

00000400 00000200 [ SachkoCronusBirthBuffer db 512 dup (?)

00

]

00000600 00000080 [ SachkoHestiaABuffer db 128 dup(?)

00

]

00000680 00000080 [ SachkoDemeterBBuffer db 128 dup(?)

00

]

00000700 00000080 [ SachkoHadesCBuffer db 128 dup(?)

00

]

00000780 00000100 [ SachkoPoseidonDBuffer db 256 dup(?)

00

]

00000880 00000100 [ SachkoHeraEBuffer db 256 dup(?)

00

]

00000980 00000100 [ SachkoZeusFBuffer db 256 dup(?)

00

]

00000A80 00000100 [ SachkoPoseidonEuphrosynePositiveDBuffer db 256 dup(?)

00

]

00000B80 00000200 [ SachkoPoseidonOizysNegativeDBuffer db 512 dup(?)

00

]

00000D80 00000100 [ SachkoHeraEuphrosynePositiveEBuffer db 256 dup(?)

00

]

00000E80 00000200 [ SachkoHeraOizysNegativeEBuffer db 512 dup(?)

00

]

00001080 00000100 [ SachkoZeusEuphrosynePositiveFBuffer db 256 dup(?)

00

]

00001180 00000200 [ SachkoZeusOizysNegativeFBuffer db 512 dup(?)

00

]

00000000 .data

; Date of birth character string

00000000 **30 37 30 34 32** SachkoCronusDateOfBirth db "07042005",0

**30 30 35** 00

; Format: Byte

00000009 **07** SachkoHestiaEuphrosynePositiveANumberByte db 07

0000000A **F9** SachkoHestiaOizysNegativeANumberByte db -07

; Format: Word

0000000B **0007** SachkoHestiaEuphrosynePositiveANumberWord dw 07

0000000D **FFF9** SachkoHestiaOizysNegativeANumberWord dw -07

0000000F **02C0** SachkoDemeterEuphrosynePositiveBNumberWord dw 0704

00000011 **FD40** SachkoDemeterOizysNegativeBNumberWord dw -0704

; Format: ShortInt

00000013 **00000007** SachkoHestiaEuphrosynePositiveANumberShortInt dd 07

00000017 **FFFFFFF9** SachkoHestiaOizysNegativeANumberShortInt dd -07

0000001B **000002C0** SachkoDemeterEuphrosynePositiveBNumberShortInt dd 0704

0000001F **FFFFFD40** SachkoDemeterOizysNegativeBNumberShortInt dd -0704

00000023 **006B73D5** SachkoHadesEuphrosynePositiveCNumberShortInt dd 07042005

00000027 **FF948C2B** SachkoHadesOizysNegativeCNumberShortInt dd -07042005

; Format: LongInt

0000002B SachkoHestiaEuphrosynePositiveANumberLongInt dq 07

**0000000000000007**

00000033 SachkoHestiaOizysNegativeANumberLongInt dq -07

**FFFFFFFFFFFFFFF9**

0000003B SachkoDemeterEuphrosynePositiveBNumberLongInt dq 0704

**00000000000002C0**

00000043 SachkoDemeterOizysNegativeBNumberLongInt dq -0704

**FFFFFFFFFFFFFD40**

0000004B SachkoHadesEuphrosynePositiveCNumberLongInt dq 07042005

**00000000006B73D5**

00000053 SachkoHadesOizysNegativeCNumberLongInt dq -07042005

**FFFFFFFFFF948C2B**

; Format: Single (float)

0000005B **3A83126F** SachkoPoseidonEuphrosynePositiveDNumberSingleFloat dd 0.001

0000005F **BA83126F** SachkoPoseidonOizysNegativeDNumberSingleFloat dd -0.001

; Format: Double (double)

00000063 SachkoHeraEuphrosynePositiveENumberDoubelDouble dq 0.077

**3FB3B645A1CAC083**

0000006B SachkoHeraOizysNegativeENumberDoubleDouble dq -0.077

**BFB3B645A1CAC083**

00000073 SachkoPoseidonEuphrosynePositiveDNumberDoubelDouble dq 0.001

3F50624DD2F1A9FC

0000007B SachkoPoseidonOizysNegativeDNumberDoubleDouble dq -0.001

BF50624DD2F1A9FC

00000083 SachkoZeusEuphrosynePositiveFNumberDoubelDouble dq 769.786

40880E49BA5E353F

0000008B SachkoZeusOizysNegativeFNumberDoubleDouble dq -769.786

C0880E49BA5E353F

; Format: Extended (long double)

00000093 SachkoZeusEuphrosynePositiveFNumberExtendetLongDouble dt 769.786

**4008C0724DD2F1A9FBE7**

0000009D SachkoZeusOizysNegativeFNumberExtendetLongDouble dt -769.786

**C008C0724DD2F1A9FBE7**

000000A7 4D 79 20 64 61 SachkoCronusDateOfBirthForma db "My date of birth (Sachko): %s",0

74 65 20 6F

66 20 62 69

72 74 68 20

28 53 61 63

68 6B 6F 29

3A 20 25 73

00

000000C5 41 20 6E 75 6D SachkoHestiaAForma db "A numbers (day of birth):",10,

62 65 72 73

20 28 64 61

79 20 6F 66

20 62 69 72

74 68 29 3A

0A 41 20 3D

20 25 64 09

09 2D 41 20

3D 20 25 64

00

"A = %d",9,9,"-A = %d",0

000000EF 42 20 6E 75 6D SachkoDemeterBForma db "B numbers (day and month of birth):",10,

62 65 72 73

20 28 64 61

79 20 61 6E

64 20 6D 6F

6E 74 68 20

6F 66 20 62

69 72 74 68

29 3A 0A 42

20 3D 20 25

64 09 09 2D

42 20 3D 20

25 64 00

"B = %d",9,9,"-B = %d",0

00000123 43 20 6E 75 6D SachkoHadesCForma db "C numbers (full date of birth):",10,

62 65 72 73

20 28 66 75

6C 6C 20 64

61 74 65 20

6F 66 20 62

69 72 74 68

29 3A 0A 43

20 3D 20 25

64 09 2D 43

20 3D 20 25

64 00

"C = %d",9,"-C = %d",0

00000152 44 20 6E 75 6D SachkoPoseidonDForma db "D numbers (A divided by the number of the score book):",10,

62 65 72 73

20 28 41 20

64 69 76 69

64 65 64 20

62 79 20 74

68 65 20 6E

75 6D 62 65

72 20 6F 66

20 74 68 65

20 73 63 6F

72 65 20 62

6F 6F 6B 29

3A 0A 44 20

3D 20 25 73

09 2D 44 20

3D 20 25 73

00

"D = %s",9,"-D = %s",0

00000198 45 20 6E 75 6D SachkoHeraEForma db "E numbers (B divided by the number of the score book):",10,

62 65 72 73

20 28 42 20

64 69 76 69

64 65 64 20

62 79 20 74

68 65 20 6E

75 6D 62 65

72 20 6F 66

20 74 68 65

20 73 63 6F

72 65 20 62

6F 6F 6B 29

3A 0A 45 20

3D 20 25 73

09 09 2D 45

20 3D 20 25

73 00

"E = %s",9,9,"-E = %s",0

000001DF 46 20 6E 75 6D SachkoZeusFForma db "F numbers (C divided by the number of the score book):",10,

62 65 72 73

20 28 43 20

64 69 76 69

64 65 64 20

62 79 20 74

68 65 20 6E

75 6D 62 65

72 20 6F 66

20 74 68 65

20 73 63 6F

72 65 20 62

6F 6F 6B 29

3A 0A 46 20

3D 20 25 73

09 2D 46 20

3D 20 25 73

00

"F = %s",9,"-F = %s",0

00000225 25 73 0A 4D 79 SachkoMnemosyneBodyToDisplayInformationInADialogBox db "%s",10,

20 6E 75 6D

62 65 72 20

6F 66 20 74

68 65 20 73

63 6F 72 65

20 62 6F 6F

6B 3A 20 39

31 34 38 0A

0A 25 73 0A

0A 25 73 0A

0A 25 73 0A

0A 25 73 0A

0A 25 73 0A

0A 25 73 0A

00

"My number of the score book: 9148",10,10,

"%s",10,10,

"%s",10,10,

"%s",10,10,

"%s",10,10,

"%s",10,10,

"%s",10,0

00000263 54 68 65 20 77 SachkoCaptionOlympusDialogBoxTitle db "The work of student from the IM-22 group Maksym Sachko (option 20)",0

6F 72 6B 20

6F 66 20 73

74 75 64 65

6E 74 20 66

72 6F 6D 20

74 68 65 20

49 4D 2D 32

32 20 67 72

6F 75 70 20

4D 61 6B 73

79 6D 20 53

61 63 68 6B

6F 20 28 6F

70 74 69 6F

6E 20 32 30

29 00

00000000 .code

00000000 SachkoSecondLabTitan:

invoke FloatToStr2, SachkoPoseidonEuphrosynePositiveDNumberDoubelDouble, addr SachkoPoseidonEuphrosynePositiveDBuffer

invoke FloatToStr2, SachkoPoseidonOizysNegativeDNumberDoubleDouble, addr SachkoPoseidonOizysNegativeDBuffer

invoke FloatToStr2, SachkoHeraEuphrosynePositiveENumberDoubelDouble, addr SachkoHeraEuphrosynePositiveEBuffer

invoke FloatToStr2, SachkoHeraOizysNegativeENumberDoubleDouble, addr SachkoHeraOizysNegativeEBuffer

invoke FloatToStr, SachkoZeusEuphrosynePositiveFNumberDoubelDouble, addr SachkoZeusEuphrosynePositiveFBuffer

invoke FloatToStr, SachkoZeusOizysNegativeFNumberDoubleDouble, addr SachkoZeusOizysNegativeFBuffer

invoke wsprintf, addr SachkoCronusBirthBuffer, addr SachkoCronusDateOfBirthForma, addr SachkoCronusDateOfBirth

invoke wsprintf, addr SachkoHestiaABuffer, addr SachkoHestiaAForma, SachkoHestiaEuphrosynePositiveANumberShortInt, SachkoHestiaOizysNegativeANumberShortInt

invoke wsprintf, addr SachkoDemeterBBuffer, addr SachkoDemeterBForma, SachkoDemeterEuphrosynePositiveBNumberShortInt, SachkoDemeterOizysNegativeBNumberShortInt

invoke wsprintf, addr SachkoHadesCBuffer, addr SachkoHadesCForma, SachkoHadesEuphrosynePositiveCNumberShortInt, SachkoHadesOizysNegativeCNumberShortInt

invoke wsprintf, addr SachkoPoseidonDBuffer, addr SachkoPoseidonDForma, addr SachkoPoseidonEuphrosynePositiveDBuffer, addr SachkoPoseidonOizysNegativeDBuffer

invoke wsprintf, addr SachkoHeraEBuffer, addr SachkoHeraEForma, addr SachkoHeraEuphrosynePositiveEBuffer, addr SachkoHeraOizysNegativeEBuffer

invoke wsprintf, addr SachkoZeusFBuffer, addr SachkoZeusFForma, addr SachkoZeusEuphrosynePositiveFBuffer, addr SachkoZeusOizysNegativeFBuffer

invoke wsprintf, addr SachkoOlimpusFinalBufferForAll,

addr SachkoMnemosyneBodyToDisplayInformationInADialogBox,

addr SachkoCronusBirthBuffer,

addr SachkoHestiaABuffer,

addr SachkoDemeterBBuffer,

addr SachkoHadesCBuffer,

addr SachkoPoseidonDBuffer,

addr SachkoHeraEBuffer,

addr SachkoZeusFBuffer

invoke MessageBox, 0, addr SachkoOlimpusFinalBufferForAll, addr SachkoCaptionOlympusDialogBoxTitle, 0

invoke ExitProcess, 0

end SachkoSecondLabTitan

**12. Порівняти результати, що містяться в лістингу, з розрахунковими, отриманими при виконанні пунктів 3 – 6, і зробити висновки по лабораторній роботі**

|  |  |  |  |
| --- | --- | --- | --- |
| Константа | Дані зі звіту | Дані з лістингу | Чи співпадають дані |
| “ddmmyyyy” | ‘0’ = 3016  ‘7’ = 3716  ‘0’ = 3016  ‘4’ = 3416  ‘2’ = 3216  ‘0’ = 3016  ‘0’ = 3016  ‘5’ = 3516 | **30 37 30 34 32 30 30 35** 00 | Так |
| A (однобайтовий формат Byte) | 0716 | **07** SachkoHestiaEuphrosynePositiveANumberByte db 07 | Так |
| -A (однобайтовий формат Byte) | F916 | **F9** SachkoHestiaOizysNegativeANumberByte db -07 | Так |
| A (двобайтовий формат Word) | 000716 | **0007** SachkoHestiaEuphrosynePositiveANumberWord dw 07 | Так |
| -A (двобайтовий формат Word) | FFF916 | **FFF9** SachkoHestiaOizysNegativeANumberWord dw -07 | Так |
| B (двобайтовий формат Word) | 02C016 | **02C0** SachkoDemeterEuphrosynePositiveBNumberWord dw 0704 | Так |
| -B (двобайтовий формат Word) | FD4016 | **FD40** SachkoDemeterOizysNegativeBNumberWord dw -0704 | Так |
| A (чотирьохбайтовий формат Shortlnt) | 00 00 00 0716 | **00000007** SachkoHestiaEuphrosynePositiveANumberShortInt dd 07 | Так |
| -A (чотирьохбайтовий формат Shortlnt) | FF FF FF F916 | **FFFFFFF9** SachkoHestiaOizysNegativeANumberShortInt dd -07 | Так |
| B (чотирьохбайтовий формат Shortlnt) | 00 00 02 C016 | **000002C0** SachkoDemeterEuphrosynePositiveBNumberShortInt dd 0704 | Так |
| -B (чотирьохбайтовий формат Shortlnt) | FF FF FD 4016 | **FFFFFD40** SachkoDemeterOizysNegativeBNumberShortInt dd -0704 | Так |
| C (чотирьохбайтовий формат Shortlnt) | 00 6B 73 D516 | **006B73D5** SachkoHadesEuphrosynePositiveCNumberShortInt dd 07042005 | Так |
| -C (чотирьохбайтовий формат Shortlnt) | FF 94 8C 2B16 | **FF948C2B** SachkoHadesOizysNegativeCNumberShortInt dd -07042005 | Так |
| A (восьмибайтовий формат Longlnt) | 00 00 00 00 00 00 00 0716 | SachkoHestiaEuphrosynePositiveANumberLongInt dq 07  **0000000000000007** | Так |
| -A (восьмибайтовий формат Longlnt) | FF FF FF FF FF FF FF F916 | SachkoHestiaOizysNegativeANumberLongInt dq -07  **FFFFFFFFFFFFFFF9** | Так |
| B (восьмибайтовий формат Longlnt) | 00 00 00 00 00 00 02 C016 | SachkoDemeterEuphrosynePositiveBNumberLongInt dq 0704  **00000000000002C0** | Так |
| -B (восьмибайтовий формат Longlnt) | FF FF FF FF FF FF FD 4016 | SachkoDemeterOizysNegativeBNumberLongInt dq -0704  **FFFFFFFFFFFFFD40** | Так |
| C (восьмибайтовий формат Longlnt) | 00 00 00 00 00 6B 73 D516 | SachkoHadesEuphrosynePositiveCNumberLongInt dq 07042005  **00000000006B73D5** | Так |
| -C (восьмибайтовий формат Longlnt) | FF FF FF FF FF 94 8C 2B16 | SachkoHadesOizysNegativeCNumberLongInt dq -07042005  **FFFFFFFFFF948C2B** | Так |

|  |  |  |  |
| --- | --- | --- | --- |
| D (Чотирьохбайтовий формат Single(float)) | 3A 80 00 0016 | **3A83126F** SachkoPoseidonEuphrosynePositiveDNumberSingleFloat dd 0.001 | Співпадають перші три цифри, бо ми округлювали до 10 знаків після коми |
| -D  (Чотирьохбайтовий формат Single(float)) | BA 80 00 0016 | **BA83126F** SachkoPoseidonOizysNegativeDNumberSingleFloat dd -0.001 | Співпадають перші три цифри, бо ми округлювали до 10 знаків після коми |
| E  (Восьмибайтовий формат Double(double)) | 3F B3 C0 00 00 00 00 0016 | SachkoHeraEuphrosynePositiveENumberDoubelDouble dq 0.077  **3FB3B645A1CAC083** | Співпадають перші чотири цифри, бо ми округлювали до 10 знаків після коми |
| -E  (Восьмибайтовий формат Double(double)) | BF B3 C0 00 00 00 00 0016 | SachkoHeraOizysNegativeENumberDoubleDouble dq -0.077  **BFB3B645A1CAC083** | Співпадають перші чотири цифри, бо ми округлювали до 10 знаків після коми |
| F (Десятибайтовий формат Extended (long double)) | 40 08 C0 72 50 00 00 00 00 0016 | SachkoZeusEuphrosynePositiveFNumberExtendetLongDouble dt 769.786  **4008C0724DD2F1A9FBE7** | Співпадають перші вісім цифр, бо ми округлювали до 10 знаків після коми |
| -F  (Десятибайтовий формат Extended (long double)) | C0 08 C0 72 50 00 00 00 00 0016 | SachkoZeusOizysNegativeFNumberExtendetLongDouble dt -769.786  **C008C0724DD2F1A9FBE7** | Співпадають перші вісім цифр, бо ми округлювали до 10 знаків після коми |

**Висновок:** У цій лабораторній роботі я виконав представлення цілих і дійсних чисел у десятковій, двійковій та шістнадцядковій системах числення й у двобайтовому, чотирьохбайтовому, восьмибайтовому та навіть десятибайтовому форматах вручну та за допомогою мови Assembler MASM32.

Отримав знання методів представлення та переведення чисел із одної системи числення в іншу. Також ознайомився із представленням символів у шістнадцядкову систему числення.

Освоїв базові знання роботи з ініціалізацією змінних різних типів і форматів та роботи з адресами змінних. Також порівняв результати представлення цілих і дійсних чисел, отриманих теоретично, з результатами, що містяться в отриманому розширеному лістингу програми.

Після порівняння я дійшов до висновку, що результати теоретичних розрахунків та лістингу співпадають, однак у чисел з плаваючою комою не все збігається, оскільки ми округлювали значення до 10 знаків після коми. Тому варто зазначити, що точність теоретично розрахованих дробових чисел залежить від кількості знаків після коми, на які ми округлюємо їх при переведенні в двійкову систему числення.