**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 №1**

з дисципліни  
«Технології розроблення системних програм»

на тему  
«Внутрішнє представлення цілих і дійсних даних у процесорі ix86»

Виконала: Перевірив:

студентка групи ІП-93 доц. кафедри ОТ  
Узунлу Ґамзенур Павлов В. Г.  
номер залікової книжки: 9882

Київ 2020

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

* **A** – ціле двозначне число, що дорівнює **dd**;
* **B** – ціле чотиризначне число, яке дорівнює **ddmm**;
* **C** – ціле восьмизначне число, яке дорівнює **ddmmyyyy**;
* **D** – число, отримане шляхом ділення числа **A** на число **N**, де **N** - чотиризначний номер залікової книжки;
* **E** – число, отримане шляхом ділення числа **B** на число **N**;
* **F** - число, отримане шляхом ділення числа **C** на число **N**;

*(при діленні округлювати результат до 3 знаків після коми)*

**A** = 11

**B** = 1110

**C** = 11101999

**N** = 9882

**D** = 0.001  
**E** = 0.112

**F** = 1123.456

**2. Переведення у двійковий та шістнадцятковий коди:**

**A** =

**B** =

**C** =

|  |  |  |
| --- | --- | --- |
| **Число** | **div 16** | **mod 16** |
| 12092001 | 693875 | 1 |
| 693875 | 43367 | 6 |
| 43367 | 2710 | 2 |
| 2710 | 169 | 8 |
| 169 | 10 | 8 |
| 10 | 0 | B |

**D** = **E** =

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Число** | **Ціла частина** |  | **Число** | **Ціла частина** |
| 0.001 \* 2 | 0 |  | 0.112 \* 2 | 0 |
| 0.002 \* 2 | 0 |  | 0.224 \* 2 | 0 |
| 0.004 \* 2 | 0 |  | 0.448 \* 2 | 1 |
| 0.016 \* 2 | 0 |  | 0.024 \* 2 | 0 |
| 0.032 \* 2 | 0 |  | 0.048 \* 2 | 0 |
| 0.064 \* 2 | 0 |  | 0.096 \* 2 | 0 |
| 0.128 \* 2 | 0 |  | 0.192 \* 2 | 0 |
| 0.256 \* 2 | 0 |  | 0.384 \* 2 | 0 |
| 0.512 \* 2 | 1 |  | 0.768 \* 2 | 1 |
| 0.024 \* 2 | 0 |  | 0.632 \* 2 | 1 |

**F** =

|  |  |  |  |
| --- | --- | --- | --- |
| **Число** | **Ціла частина** | **Число** | **Ціла частина** |
| 0.456 \* 2 | 1 | 0.496 \* 2 | 1 |
| 0.032 \* 2 | 0 | 0.464 \* 2 | 1 |
| 0.064 \* 2 | 0 | 0.400 \* 2 | 1 |
| 0.128 \* 2 | 0 | 0.272 \* 2 | 1 |
| 0.256 \* 2 | 0 | 0.232 \* 2 | 0 |
| 0.512 \* 2 | 1 |  |  |

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

* **“ddmmyyy”** у вигляді символьного рядка;
* числа **A** и **–A** у однобайтовому форматі **Byte**;
* числа **A**, **B**, **-A** и **-B** у двобайтовому формат **Word**;
* числа **A**, **B**, **C**, **-A**, **-B** і **-C** у чотирьохбайтовому форматі **Shortlnt**;
* числа **A**, **B**, **C**, **-A**, **-B** і **-C** у восьмибайтовому форматі **Longlnt**;

**Символьний рядок:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Символ** | 1 | 2 | 0 | 9 | 2 | 0 | 0 |
| **Код** | 31 | 32 | 30 | 39 | 32 | 30 | 30 |

**1-байтовий формат Byte:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **1 біт** | **7 біт** | | **Hex** |
| **+А** | **0** | 000 | 1100 | 0C |
| **-А** | **1** | 111 | 0100 | F4 |

**2-байтовий формат Word:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **1 біт** | **7 біт** | | **8 біт** | | **Hex** |
| **+А** | **0** | 000 | 0000 | 0000 | 1100 | 00 0C |
| **-А** | **1** | 111 | 1111 | 1110 | 0100 | FF F4 |
| **+B** | **0** | 000 | 0100 | 1011 | 1001 | 04 B9 |
| **-B** | **1** | 111 | 1011 | 0100 | 0111 | FB 47 |

**4-байтовий формат ShortInt:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **1 біт** | **7 біт** | | **8 біт** | | **8 біт** | | **8 біт** | | **Hex** |
| **+A** | **0** | 000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 1100 | 00 00 00 0С |
| **-A** | **1** | 111 | 1111 | 1111 | 1111 | 1111 | 1111 | 1111 | 0100 | FF FF FF F4 |
| **+B** | **0** | 000 | 0000 | 0000 | 0000 | 0000 | 0100 | 1011 | 1001 | 00 00 04 B9 |
| **-B** | **1** | 111 | 1111 | 1111 | 1111 | 1111 | 1011 | 0100 | 0111 | FF FF FB 47 |
| **+C** | **0** | 000 | 0000 | 1011 | 1000 | 1000 | 0010 | 0110 | 0001 | 00 B8 82 61 |
| **-C** | **1** | 111 | 1111 | 0100 | 0111 | 0111 | 1101 | 1001 | 1111 | FF 47 7D 9F |

**8-байтовий формат LongInt:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Знак** | **7 біт** | **6 байт** | **1 байт** | **Hex** |
| **+A** | **0** | 000 0000 | 0000 0000 \* 6 | 0000 1100 | 00\*7 0C |
| **-A** | **1** | 111 1111 | 1111 1111 \* 6 | 1111 0100 | FF\*7 F4 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Знак** | **7 біт** | **5 байт** | **1 байт** | **1 байт** | **Hex** |
| **+B** | **0** | 000 0000 | 0000 0000 \* 5 | 0000 0100 | 1011 1001 | 00\*6 04 B9 |
| **-B** | **1** | 111 1111 | 1111 1111 \* 5 | 1111 1011 | 0100 0111 | FF\*6 FB 47 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Знак** | **7 біт** | **4 байти** | **1 байт** | **1 байт** | **1 байт** | **Hex** |
| **+C** | **0** | 000 0000 | 0000 0000 \* 4 | 1011 1000 | 1000 0010 | 0110 0001 | 00\*5 B8 82 61 |
| **-C** | **1** | 111 1111 | 1111 1111 \* 4 | 0100 0111 | 0111 1101 | 1001 1111 | FF\*5 87 7D 9F |

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

**D:  
E:   
F:**

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

* **D** і **–D** - у 4-байтовому форматі **Single (float)**;
* **E** і **–E** - у 8-байтовому форматі **Double (double)**;
* **F** і **–F** - у 10-байтовому форматі **Extended (long double)**.

**4-байтовий формат Single (float):**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **1 біт** | **8 біт** | **23 біти** | **IEEE 754** |
| **+D** | **0** | 0111 0101 | 000 0000 0000 0000 0000 0000 | 3A 80 00 00 |
| **-D** | **1** | 0111 0101 | 000 0000 0000 0000 0000 0000 | BA 80 00 00 |

**8-байтовий формат Double (double):**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **1 біт** | **11 біт** | **52 біти** | **IEEE 754** |
| **+E** | **0** | 011 1111 1100 | 0010 0110 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 | 3FC2600000000000 |
| **-E** | **1** | 011 1111 1100 | 0010 0110 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 | BFC2600000000000 |

**10-байтовий формат Extended (long double):**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **1 біт** | **15 біт** | **64 біти** | **IEEE 754** |
| **+F** | **0** | 100 0000 0000 1001 | 1011 0011 1011 1001 1111 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 | 4009B3B9F80000000000 |
| **-F** | **1** | 100 0000 0000 1001 | 1011 0011 1011 1001 1111 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 | C009B3B9F80000000000 |

**Вихідний код**

.486

.model flat, stdcall

option casemap: none

include /masm32/include/windows.inc

include /masm32/include/user32.inc

include /masm32/include/kernel32.inc

includelib /masm32/lib/user32.lib

includelib /masm32/lib/kernel32.lib

.data

wind\_title DB "Lab 1 Uzunlu", 0

symbol DB "1110199", 0

; Microsoft (R) Macro Assembler - Programmer's Guide Version 6.0

; https://www.pcjs.org/documents/books/mspl13/masm/mpguide/

;

; 20.3.5 The QuickPascal/MASM Interface

;

; There's information that types, such as LongInt, ShortInt have

; the following representation in MASM:

; ShortInt - SBYTE (BYTE)

; LongInt - SDWORD (DWORD)

; Single - REAL4

; Double - REAL8

; Extended - REAL10

; b - byte

; bn - byte, negative

; w - word

; wn - word, negative

; s - shortint

; sn - shortint, negative

; l = longint

; ln = longint, negative

a\_b BYTE 11

a\_bn BYTE -11

a\_w WORD 11

a\_wn WORD -11

a\_s BYTE 11

a\_sn BYTE -11

a\_l DWORD 11

a\_ln DWORD -11

b\_w WORD 1110

b\_wn WORD -1110

; BYTE can hold values of 0..255 (-128..127),

; so B can't be held as ShortInt

;; b\_s BYTE 1110

;; b\_sn BYTE -1110

b\_l DWORD 1110

b\_ln DWORD -1110

; BYTE can hold values of 0..255 (-128..127),

; so B can't be held as ShortInt

;; c\_s BYTE 11101999

;; c\_sn BYTE -11101999

c\_l DWORD 11101999

c\_ln DWORD -11101999

; s - Single

; sn - Single, negative

d\_s REAL4 0.001

d\_sn REAL4 -0.001

; d - Double

; dn - Double, negative

e\_d REAL8 0.112

e\_dn REAL8 -0.112

; e - Extended

; en - Extended, negative

f\_e REAL10 1123.456

f\_en REAL10 -1123.456

; %s - there must be string

; %x - there must be value in hexadecimal system

string DB "Symbol: '%s'", 13,

"B A: %x", 13,

"B -A: %x", 13,

"W A: %x", 13,

"W B: %x", 13,

"W -A: %x", 13,

"W -B: %x", 13,

"SI A: %x", 13,

"SI -A: %x", 13,

"LI A: %x", 13,

"LI B: %x", 13,

"LI C: %x", 13,

"LI -A: %x", 13,

"LI -B: %x", 13,

"LI -C: %x", 13,

"Sgl D: %x", 13,

"Sgl -D: %x", 13,

"Dbl E: %x", 13,

"Dbl -E: %x", 13,

"Ext F: %x", 13,

"Ext -F: %x", 0

buffer DB 2048 DUP (?)

.code

start:

invoke wsprintf, addr buffer, addr string, addr symbol,

a\_b, a\_bn, a\_w, b\_w, a\_wn, b\_wn,

a\_s, a\_sn,

a\_l, b\_l, c\_l, a\_ln, b\_ln, c\_ln,

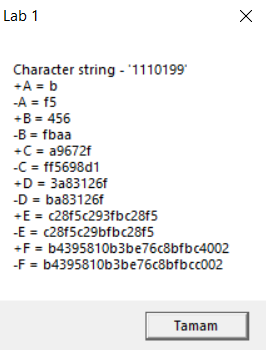
d\_s, d\_sn, e\_d, e\_dn, f\_e, f\_en

invoke MessageBox, 0, addr buffer, addr wind\_title, 0

invoke ExitProcess, 0

end start

**Скріншоти**



**Лістинг**

00000000 .data

00000000 CB E0 E1 E0 20 msg\_title DB "Lab 1", 0

31 00

00000007 D1 E8 EC E2 EE format\_symbol DB "Character string - '%s'", 0

EB FC ED E8

E9 20 F0 FF

E4 EE EA 20

2D 20 27 25

73 27 00

0000001F 00000080 [ buffer\_symbol DB 128 DUP (?)

00

]

0000009F 31 32 30 39 32 symbol DB "110199", 0

30 30 00

000000A7 2B 41 20 3D 20 format\_a\_pos DB "+A = %x", 0

25 78 00

000000AF 2D 41 20 3D 20 format\_a\_neg DB "-A = %x", 0

25 78 00

000000B7 2B 42 20 3D 20 format\_b\_pos DB "+B = %x", 0

25 78 00

000000BF 2D 42 20 3D 20 format\_b\_neg DB "-B = %x", 0

25 78 00

000000C7 2B 43 20 3D 20 format\_c\_pos DB "+C = %x", 0

25 78 00

000000CF 2D 43 20 3D 20 format\_c\_neg DB "-C = %x", 0

25 78 00

000000D7 2B 44 20 3D 20 format\_d\_pos DB "+D = %x", 0

25 78 00

000000DF 2D 44 20 3D 20 format\_d\_neg DB "-D = %x", 0

25 78 00

000000E7 2B 45 20 3D 20 format\_e\_pos DB "+E = %x%x", 0

25 78 25 78

00

000000F1 2D 45 20 3D 20 format\_e\_neg DB "-E = %x%x", 0

25 78 25 78

00

000000FB 2B 46 20 3D 20 format\_f\_pos DB "+F = %x%x%x", 0

25 78 25 78

25 78 00

00000107 2D 46 20 3D 20 format\_f\_neg DB "-F = %x%x%x", 0

25 78 25 78

25 78 00

00000113 00000010 [ buffer\_a\_pos DB 16 DUP (?)

00

]

00000123 00000010 [ buffer\_a\_neg DB 16 DUP (?)

00

]

00000133 00000010 [ buffer\_b\_pos DB 16 DUP (?)

00

]

00000143 00000010 [ buffer\_b\_neg DB 16 DUP (?)

00

]

00000153 00000010 [ buffer\_c\_pos DB 16 DUP (?)

00

]

00000163 00000010 [ buffer\_c\_neg DB 16 DUP (?)

00

]

00000173 00000010 [ buffer\_d\_pos DB 16 DUP (?)

00

]

00000183 00000010 [ buffer\_d\_neg DB 16 DUP (?)

00

]

00000193 00000020 [ buffer\_e\_pos DB 32 DUP (?)

00

]

000001B3 00000020 [ buffer\_e\_neg DB 32 DUP (?)

00

]

000001D3 00000020 [ buffer\_f\_pos DB 32 DUP (?)

00

]

000001F3 00000020 [ buffer\_f\_neg DB 32 DUP (?)

00

]

00000213 25 73 0D 25 73 msg\_format DB "%s", 13,

0D 25 73 0D

25 73 0D 25

73 0D 25 73

0D 25 73 0D

25 73 0D 25

73 0D 25 73

0D 25 73 0D

25 73 0D 25

73 0D 00

"%s", 13, ; +A

"%s", 13, ; -A

"%s", 13, ; +B

"%s", 13, ; -B

"%s", 13, ; +C

"%s", 13, ; -C

"%s", 13, ; +D

"%s", 13, ; -D

"%s", 13, ; +E

"%s", 13, ; -E

"%s", 13, ; +F

"%s", 13, ; -F

0

0000023B 00000080 [ msg\_buffer DB 128 DUP (?)

00

]

; BYTE

000002BB 0C A\_byte\_pos DB +11

000002BC F4 A\_byte\_neg DB -11

; WORD

000002BD 000C A\_word\_pos DW +11

000002BF FFF4 A\_word\_neg DW -11

000002C1 04B9 B\_word\_pos DW +1110

000002C3 FB47 B\_word\_neg DW -1110

; SHORT INT

000002C5 0000000C A\_short\_pos DD +11

000002C9 FFFFFFF4 A\_short\_neg DD -11

000002CD 000004B9 B\_short\_pos DD +1110

000002D1 FFFFFB47 B\_short\_neg DD -1110

000002D5 00B88261 C\_short\_pos DD +11101999

000002D9 FF477D9F C\_short\_neg DD -11101999

; LONG INT

000002DD A\_long\_pos DQ +11

000000000000000C

000002E5 A\_long\_neg DQ -11

FFFFFFFFFFFFFFF4

000002ED B\_long\_pos DQ +1110

00000000000004B9

000002F5 B\_long\_neg DQ -1110

FFFFFFFFFFFFFB47

000002FD C\_long\_pos DQ +11101999

0000000000B88261

00000305 C\_long\_neg DQ -11101999

FFFFFFFFFF477D9F

; FLOAT

0000030D 3A83126F D\_float\_pos DD +0.001

00000311 BA83126F D\_float\_neg DD -0.001

; DOUBLE

00000315 E\_double\_pos DQ +0.112

3FC26E978D4FDF3B

0000031D E\_double\_neg DQ -0.112

BFC26E978D4FDF3B

; LONG DOUBLE

00000325 F\_longD\_pos DT +1123.456

4009B3B9FBE76C8B4396

0000032F F\_longD\_neg DT -1123.456

C009B3B9FBE76C8B4396

00000000 .code

00000000 start:

invoke wsprintf, addr buffer\_symbol, addr format\_symbol, addr symbol

invoke wsprintf, addr buffer\_a\_pos, addr format\_a\_pos, A\_byte\_pos

invoke wsprintf, addr buffer\_a\_neg, addr format\_a\_neg, A\_byte\_neg

invoke wsprintf, addr buffer\_b\_pos, addr format\_b\_pos, B\_word\_pos

invoke wsprintf, addr buffer\_b\_neg, addr format\_b\_neg, B\_word\_neg

invoke wsprintf, addr buffer\_c\_pos, addr format\_c\_pos, C\_short\_pos

invoke wsprintf, addr buffer\_c\_neg, addr format\_c\_neg, C\_short\_neg

invoke wsprintf, addr buffer\_d\_pos, addr format\_d\_pos, D\_float\_pos

invoke wsprintf, addr buffer\_d\_neg, addr format\_d\_neg, D\_float\_neg

invoke wsprintf, addr buffer\_e\_pos, addr format\_e\_pos, E\_double\_pos

invoke wsprintf, addr buffer\_e\_neg, addr format\_e\_neg, E\_double\_neg

invoke wsprintf, addr buffer\_f\_pos, addr format\_f\_pos, F\_longD\_pos

invoke wsprintf, addr buffer\_f\_neg, addr format\_f\_neg, F\_longD\_neg

invoke wsprintf, addr msg\_buffer, addr msg\_format,

addr buffer\_symbol,

addr buffer\_a\_pos,

addr buffer\_a\_neg,

addr buffer\_b\_pos,

addr buffer\_b\_neg,

addr buffer\_c\_pos,

addr buffer\_c\_neg,

addr buffer\_d\_pos,

addr buffer\_d\_neg,

addr buffer\_e\_pos,

addr buffer\_e\_neg,

addr buffer\_f\_pos,

addr buffer\_f\_neg

invoke MessageBox, 0, addr msg\_buffer, addr msg\_title, MB\_OK

invoke ExitProcess, 0

end start

**Висновки**

Результати

Після розробки програми ми переконались, що комп’ютер використовує формат IEEE 754 для зберігання дробових чисел. Результати розрахунків та скріншоти виявилися рівними.