МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”

Лабораторна робота №3

з предмету “Системне програмування”

Виконав:

студент 2-го курсу ФІОТ

групи ІО-31

Долинний О.В.

Київ 2015

Лабораторна робота №3

Створення модульних проектів на асемблері у середовищі Visual Studio та вивчення форматів представлення чисел

Мета: Навчитися створювати модульні проекти на асемблері, а також

закріпити знання основних форматів представлення чисел у комп’ютері.

**Завдання:**

1. Створити у середовищі MS Visual Studio проект з ім’ям Lab3.

2. Написати вихідний текст програми згідно варіанту завдання. Вихідний текст

повинен бути у вигляді двох модулів на асемблері:

- головний модуль, у якому описується загальний хід виконання програми

від початку і до завершення. Цей модуль містить точку входу у програму,

впродовж роботи викликає процедури з інших модулів. Вихідний текст

головного модуля записати у файл main3.asm;

- другий модуль, який містить процедуру, яка викликається з головного

модуля. Цей модуль записати у файл module.asm.

3. Додати файли модулів у проект. У цьому проекті кожний модуль може

окремо компілюватися.

4. Скомпілювати вихідний текст і отримати виконуємий файл програми.

5. Перевірити роботу програми. Налагодити програму.

6. Отримати результати – кодовані значення чисел згідно варіанту завдання.

7. Проаналізувати та прокоментувати результати та вихідний текст.

**Варіант завдання:** N=13  
X = N+10 = 23; Y = 2×X = 46

Роздруківка тексту програми

.586

.model flat, stdcall

option casemap :none

include \masm32\include\kernel32.inc

include \masm32\include\user32.inc

include \masm32\include\windows.inc

include module.inc

includelib \masm32\lib\kernel32.lib

includelib \masm32\lib\user32.lib

.const

.data

TextBuf db 64 dup(?)

Caption db "Lab 3", 0

Value1 db 21

Value2 db -21

Value3 dw 21

Value4 dw -21

Value5 dd 21

Value6 dd -21

Value7 dq 21

Value8 dq -21

Value9 dd 21.0

Value10 dd -42.0

Value11 dd 21.21

Value12 dq 21.0

Value13 dq -42.0

Value14 dq 21.21

Value15 dt 21.0

Value16 dt -42.0

Value17 dt 21.21

.code

main:

push offset TextBuf

push offset Value1

push 8

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value2

push 8

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value3

push 16

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value4

push 16

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value5

push 32

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value6

push 32

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value7

push 64

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value8

push 64

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value9

push 32

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value10

push 32

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value11

push 32

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value12

push 64

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value13

push 64

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value14

push 64

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value15

push 80

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value16

push 80

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

push offset TextBuf

push offset Value17

push 80

call StrHex\_MY

invoke MessageBox, 0, ADDR TextBuf, ADDR Caption, MB\_ICONINFORMATION

invoke ExitProcess, 0

invoke ExitProcess, 0

end main

**Результати:**

|  |  |  |  |
| --- | --- | --- | --- |
| Тип даних | Значення | Результати виконання програми | |
| шістнадцятковий код | двійковий код |
| Ціле 8-бітове | 21 | 15 | 0001 0101 |
| -21 | EB | 1110 1011 |
| Ціле 16-бітове | 21 | 0015 | 0000 0000 0001 0101 |
| -21 | FFEB | 1111 1111 1110 1011 |
| Ціле 32-бітове | 21 | 0000 0015 | 0000 0000 0000 0000 0000 0000 0001 0101 |
| -21 | FFFF FFEB | 1111 1111 1111 1111 1111 1111 1110 1011 |
| Ціле 64-бітове | 21 | 0000 0000 0000 0015 | 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0001 0101 |
| -21 | FFFF FFFF FFFF FFEB | 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1111 1110 1011 |
| Число у 32-бітовому форматі | 21.0 | 41A8 0000 | 0100 0001 1010 1000 0000 0000 0000 0000 |
| -46.0 | C228 0000 | 1100 0010 0010 1000 0000 0000 0000 0000 |
| 21.21 | 41A9 AE14 | 0100 0001 1010 1001 1010 1101 0001 0100 |
| Число у 64-бітовому форматі  з плаваючою точкою | 21.0 | 4035 0000 0000 0000 | 0100 0000 0011 0101 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 |
| -46.0 | C045 0000 0000 0000 | 1100 0000 0100 0101 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 |
| 21.21 | 4035 35C2 8F5C 28F6 | 0100 0000 0011 0101 0011 0101 1100 0010 1000 1111 0101 1100 0010 1000 1111 0110 |
| Число у 80-бітовому форматі  з плаваючою точкою | 21.0 | 4003 A800 0000 0000 0000 | 0100 0000 0000 0011 1010 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 |
| -46.0 | C004 A800 0000 0000 0000 | 1100 0000 0000 0100 1010 1000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 |
| 21.21 | 4003 A9AE 147A E147 AE14 | 0100 0000 0000 0011 1010 1001 1010 1110 0001 0100 0111 1010 1110 0001 0100 0111 1010 1110 0001 0100 |

Аналіз результатів:

32 біти 1 біт-знак, 2-9 експонента, 10-32 мантисса

64 біти 1 біт-знак, 2-12 експонента, 13-64 мантисса

80 біт 1 біт-знак, 2-16 експонента,17-ціла част., 18-80 мантисса

**Висновок:**

Під час виконання роботи мною були вивчені основи створювання модульних програм в середовищі Microsoft Visual Studio та вивчені основні формати представлення чисел.