Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

ЛАБОРАТОРНА РОБОТА № 2

З дисципліни «Системне програмування»

ВИКОНАВ:

студент 2 курсу ФІОТ

групи ІО-41

Логвинчук А. І.

ПЕРЕВІРИВ:

ст. вик. Порєв В. М.

Київ – 2016

**Тема:** Знайомство із середовищем розробки програм Microsoft Visual Studio.

**Мета:** Отримати перші навички роботи з Microsoft Visual Studio для створення

програм, написаних мовою асемблера, а також вивчити команди MOV та

CPUID.

**Завдання:**

1. Створити у середовищі MS Visual Studio проект з ім’ям Lab2. Встановити

необхідні параметри проекту – опції середовища розробки програм.

2. Написати вихідний текст програми на асемблері, додати файл вихідного

тексту у проект. Зміст вихідного тексту згідно з варіантом завдання.

3. Скомпілювати вихідний текст і отримати виконуємий файл програми.

4. Перевірити роботу програми. Налагодити програму.

5. Отримати дизасембльований текст машинного коду і проаналізувати його.

**Програмний код**

.586

.model flat, stdcall

include \DEV\masm32\include\kernel32.inc

include \DEV\masm32\include\user32.inc

includelib \DEV\masm32\lib\kernel32.lib

includelib \DEV\masm32\lib\user32.lib

.data

TextInit db "Вивчення команд mov та cpuid", 13, 10,

"вик. Логвинчук А. І., група ІО-41", 0

Text0 db "Vendor=xxxxxxxxxxxx", 13, 10,

"MaxValue=xxxxxxxx", 0

Text80 db "MaxExtVal=xxxxxxxx", 0

Text db "EAX=xxxxxxxx", 13, 10,

"EBX=xxxxxxxx", 13, 10,

"ECX=xxxxxxxx", 13, 10,

"EDX=xxxxxxxx", 0

CaptionInit db "Лаб. робота №2", 0

Caption00 db "CPUID 00h", 0

Caption01 db "CPUID 01h", 0

Caption02 db "CPUID 02h", 0

CaptionExt db "CPUID EXT", 0

Caption81 db "CPUID 80..01h", 0

Caption82 db "CPUID 80..02h", 0

Caption83 db "CPUID 80..03h", 0

Caption84 db "CPUID 80..04h", 0

Caption85 db "CPUID 80..05h", 0

Caption88 db "CPUID 80..08h", 0

.code

DwordToStrHex proc

push ebp

mov ebp, esp

mov ebx, [ebp+8] ;другий параметр

mov edx, [ebp+12] ;перший параметр

xor eax, eax

mov edi, 7

@next:

mov al, dl

and al, 0Fh ;виділяємо одну шістнадцяткову цифру

add ax, 48 ;так можна тільки для цифр 0-9

cmp ax, 58

jl @store

add ax, 7 ;для цифр A, B, C, D, E, F

@store:

mov [ebx+edi], al

shr edx, 4

dec edi

cmp edi, 0

jge @next

pop ebp

ret 8

DwordToStrHex endp

main:

;------------------------------------Привітання-----------------------------

invoke MessageBoxA, 0, ADDR TextInit, ADDR CaptionInit, 0

;-------------------Виклик CPUID з параметром 0-----------------------------

mov eax, 00h

cpuid

mov dword ptr[Text0+7], ebx

mov dword ptr[Text0+11], edx

mov dword ptr[Text0+15], ecx

push eax

push offset [Text0+30]

call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text0, ADDR Caption00, 0

;-------------------Виклик CPUID з параметром 1-----------------------------

mov eax, 01h

cpuid

push eax

push offset[Text+4]

call DwordToStrHex

push ebx

push offset[Text+18]

call DwordToStrHex

push ecx

push offset[Text+32]

call DwordToStrHex

push edx

push offset[Text+46]

call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text, ADDR Caption01, 0

;-------------------Виклик CPUID з параметром 2-----------------------------

mov eax, 02h

cpuid

push eax

push offset[Text+4]

call DwordToStrHex

push ebx

push offset[Text+18]

call DwordToStrHex

push ecx

push offset[Text+32]

call DwordToStrHex

push edx

push offset[Text+46]

call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text, ADDR Caption02, 0

;-------------------Виклик CPUID з параметром 80000000h-----------------------------

;Extended Function CPUID Information

mov eax, 80000000h

cpuid

push eax

push offset[Text80+10]

call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text80, ADDR CaptionExt, 0

;-------------------Виклик CPUID з параметром 80000001h-----------------------------

mov eax, 80000001h

cpuid

push eax

push offset[Text+4]

call DwordToStrHex

push ebx

push offset[Text+18]

call DwordToStrHex

push ecx

push offset[Text+32]

call DwordToStrHex

push edx

push offset[Text+46]

call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text, ADDR Caption81, 0

;-------------------Виклик CPUID з параметром 80000002h-----------------------------

mov eax, 80000002h

cpuid

push eax

push offset[Text+4]

call DwordToStrHex

push ebx

push offset[Text+18]

call DwordToStrHex

push ecx

push offset[Text+32]

call DwordToStrHex

push edx

push offset[Text+46]

call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text, ADDR Caption82, 0

;-------------------Виклик CPUID з параметром 80000003h-----------------------------

mov eax, 80000003h

cpuid

push eax

push offset[Text+4]

call DwordToStrHex

push ebx

push offset[Text+18]

call DwordToStrHex

push ecx

push offset[Text+32]

call DwordToStrHex

push edx

push offset[Text+46]

call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text, ADDR Caption83, 0

;-------------------Виклик CPUID з параметром 80000004h-----------------------------

mov eax, 80000004h

cpuid

push eax

push offset[Text+4]

call DwordToStrHex

push ebx

push offset[Text+18]

call DwordToStrHex

push ecx

push offset[Text+32]

call DwordToStrHex

push edx

push offset[Text+46]

call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text, ADDR Caption84, 0

;-------------------Виклик CPUID з параметром 80000005h-----------------------------

mov eax, 80000005h

cpuid

push eax

push offset[Text+4]

call DwordToStrHex

push ebx

push offset[Text+18]

call DwordToStrHex

push ecx

push offset[Text+32]

call DwordToStrHex

push edx

push offset[Text+46]

call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text, ADDR Caption85, 0

;-------------------Виклик CPUID з параметром 80000008h-----------------------------

mov eax, 80000008h

cpuid

push eax

push offset[Text+4]

call DwordToStrHex

push ebx

push offset[Text+18]

call DwordToStrHex

push ecx

push offset[Text+32]

call DwordToStrHex

push edx

push offset[Text+46]

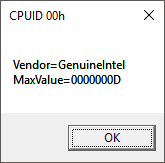
call DwordToStrHex

invoke MessageBoxA, 0, ADDR Text, ADDR Caption88, 0

invoke ExitProcess, 0

end main

**Результат виконання програми**



**Висновок**

Команда MOV використовується для пересилання даних між регістрами там пам’ятю комп’ютера. Команда CPUID надає інформацію про виробника та технічні характеристики ЦП.

Середовище розробки VisualStudio 2015 мість набір інструменітв для налагодження, серед яких є дизасемблер – засіб для перетворення машинних кодів у програму, придатну для читання людиною.