Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

ЛАБОРАТОРНА РОБОТА № 9

З дисципліни «Системне програмування»

ВИКОНАВ:

студент 2 курсу ФІОТ

групи ІО-41

Логвинчук А. І.

ПЕРЕВІРИВ:

ст. вик. Порєв В. М.

Київ – 2016

**Тема:** Використання функцій API Win32 у програмах на асемблері

**Мета:** Навчитися використовувати у програмах на асемблері функції Windows

динамічного виділення пам’яті та запису файлів.

**Варіант завдання:**

Вар. 20.

n = 70

Спосіб організації циклу: 2

Код результату: НЕХ

Масиви даних та імені файлу – динамічні.

**Програмний код**

**main9.asm:**

.586

.model flat, stdcall

option casemap: none

include D:\DEV\masm32\include\kernel32.inc

include D:\DEV\masm32\include\user32.inc

include D:\DEV\masm32\include\windows.inc

include D:\DEV\masm32\include\comdlg32.inc

include module.inc

include longop.inc

includelib D:\DEV\masm32\lib\comdlg32.lib

includelib D:\DEV\masm32\lib\kernel32.lib

includelib D:\DEV\masm32\lib\user32.lib

.data

counter dd 1

crlf dd 13, 10

wres dd ?

hFile dd 0

pVal dd ?

pRes dd ?

pFileName dd ?

TextBuf db 100 dup(0)

Caption db "70!", 0

.code

MySaveFileName proc

LOCAL ofn : OPENFILENAME

invoke RtlZeroMemory, ADDR ofn, SIZEOF ofn

mov ofn.lStructSize, SIZEOF ofn

mov esi, pFileName

mov ofn.lpstrFile, esi

mov ofn.nMaxFile, 256

invoke GetSaveFileName, ADDR ofn

ret

MySaveFileName endp

main:

invoke GlobalAlloc, GPTR, 352

mov pVal, eax

mov dword ptr[eax], 1h

add eax, 48

mov pRes, eax

add eax, 48

mov pFileName, eax

call MySaveFileName

cmp eax, 0

jz @exit

invoke CreateFile, pFileName,

GENERIC\_WRITE,

FILE\_SHARE\_WRITE,

0, CREATE\_ALWAYS,

FILE\_ATTRIBUTE\_NORMAL,

0

cmp eax, INVALID\_HANDLE\_VALUE

je @exit ; доступ до файлу неможливий

mov hFile, eax

; обчислення факторіала

@loop1:

push pRes

push pVal

push counter

call Multip\_Nx32

inc counter

xor ecx, ecx

@loop2:

mov edx, pRes

mov ebx, [edx + 4\*ecx]

mov edx, pVal

mov [edx + 4\*ecx], ebx

inc ecx

cmp ecx, 12

jle @loop2

push offset TextBuf

push pVal

push 336

call StrHex\_MY

invoke lstrlen, ADDR TextBuf

invoke WriteFile, hFile, ADDR TextBuf, eax, NULL, 0

invoke WriteFile, hFile, ADDR crlf, 1, NULL, 0

cmp counter, 70

jle @loop1

invoke CloseHandle, hFile

invoke GlobalFree, pVal

@exit:

invoke ExitProcess, 0

end main

**Результати роботи програми:**

**Висновок**

Під час виконання роботи я навчився працювати з функціями Win32 API, динамічною памяттю та створив програму, що обчислює факторіал чисел від 1 до 70 та записує їх у файл.