НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ ТА ПРОГРАМУВАННЯ»

«Системне програмування »

*Звiт з лабораторної роботи №8*

*Тема: «*Реєст,Процеси*»*

Виконав:

ст. гр. KIT-119D

Гряник Г.В.

Перевірив:

Межерицький С.Г.

Харків – 2020

**Мета :**

набути практичних навичок створення процесів, написаних мовою ассемблера з використанням АРІ-функцій.

***Індивідуальне завдання***

Варіант 6.

* Повідомлення при завантаженні, автозавантаженні. (Гілки HKLM \ Software \ Microsoft \ WindowsNT \ CurrentVersion \ Winlogon і HKLM \ SoftWare \ Microsoft \ Windows \ CurrentVersion. Ключі: LegalNoticeCaption, LegalNoticeText, RunOnce, RunOnceEx, RunServices, RunServicesOnce, RunServices, DisableLocalMachineRun, DisableLocalMachineRunOnce, DisableCurrentUserRun).
* Запуск однієї програми, створеної особисто, і програми калькулятора (calc.exe). Особисто створена програма виконується 5 секунд, а програма calc.exe виконується 8 секунд.
* Обчислити рівняння (де V - корінь) способами: за допомогою FPU (в одному потоці), за допомогою скалярних (в окремому потоці) і паралельних команд SSE (AVX) (в окремому потоці), і порівняти час їх виконання. Числа а є {a1, a2, a3, a4} задані масивом. При використанні скалярних команд змінні задані масивом {b1, c1, d1, e1}. При використанні паралельних SSE- (AVX)-команда числа b1, c1, d1, e1 уявити окремими масивами. Всі масиви розташувати в окремому зовнішньому файлі з розширенням .inc. Результати вивести через функцію MessageBoxIndirect.

**Текст програми**

Програма 1: (LR8-1)

include \masm64\include64\masm64rt.inc

include \masm64\macros64\macros64.inc

.data

hInstance dq ? ; дескриптор програми

hWnd dq ? ; дескриптор окна

hIcon dq ? ; дескриптор иконки

hCursor dq ? ; дескриптор курсора

sWid dq ? ; ширина монитора (колич. пикселей по x)

sHgt dq ? ; высота монитора (колич. пикселей по y)

title1 db "Лаб.8",0

txt01 db "Сообщение при загрузке, автозагрузке",0

txt02 db "Ветки HKLM\Software\Microsoft\WindowsNT\CurrentVersion\Winlogon",10," HKLM\ SoftWare\Microsoft\Windows\CurrentVersion. ",0

txt03 db "Ключи: LegalNoticeCaption, LegalNoticeText, RunOnce, RunOnceEx, RunServices, RunServicesOnce",0

autor db "Автор: Гряник Г.В., гр.КІТ-119Д",0

szREGSZ db 'REG\_SZ',0

hKey dd ?

lpdwDisp dd ?

.code

entry\_point proc

mov hInstance,rv(GetModuleHandle,0)

mov hIcon,rv(LoadImage,hInstance,10,IMAGE\_ICON,128,128,LR\_DEFAULTCOLOR)

; mov hBmp, rv(ResImageLoad,20,,128,128,LR\_DEFAULTCOLOR)

invoke DialogBoxParam,0,1000,0,ADDR mainW,hIcon

invoke ExitProcess,0

ret

entry\_point endp

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

mainW proc hWin:QWORD,uMsg:QWORD,wParam:QWORD,lParam:QWORD

.switch uMsg

.case WM\_INITDIALOG ; сообщение о создании диал. окна

invoke SetWindowText,hWin,title1

invoke SendMessage,hWin,WM\_SETICON,1,lParam ; отправляет сообщение окну

invoke SendMessage,rv(GetDlgItem,hWin,101),\ ; сообщение окну по дескриптору органа управления

STM\_SETIMAGE,IMAGE\_ICON,lParam

; 102 - jpg

; mov hStatic, rv(GetDlgItem,hWin,101)

; invoke SendMessage,hStatic,STM\_SETIMAGE,IMAGE\_BITMAP,hBmp

.return TRUE

.case WM\_COMMAND ; сообщение от меню или кнопки

.switch wParam

.case 103 ; кнопка

invoke MsgboxI,hWin,ADDR txt01,"Завдання",MB\_OK,10

.case 102

invoke MsgboxI,hWin,ADDR autor,"Автор",MB\_OK,10

.case 104

rcall SendMessage,hWin,WM\_SYSCOMMAND,SC\_CLOSE,0;

;/////////////////////////////////////////

.case 105 ; кнопка

invoke RegCreateKeyEx,HKEY\_LOCAL\_MACHINE,"Software\Microsoft\Windows NT\CurrentVersion\Winlogon",0,ADDR szREGSZ,REG\_OPTION\_VOLATILE,KEY\_ALL\_ACCESS,0,ADDR hKey,ADDR lpdwDisp;

cmp rax,0 ;Раздел создан успешно ?

jne m1

invoke RegSetValueEx,hKey,"LegalNoticeCaption", 0,REG\_SZ,"Hello world",15 ; размер ключа в байтах

cmp rax,0 ;ПРОВЕРКА КЛЮЧА НА СОЗДАНИЕ

jne m1

invoke MessageBox,0,chr$("Зміна пройшла успішно"),chr$("Реестр"),MB\_ICONINFORMATION

jmp m2

m1:

invoke MessageBox,0,chr$("Зміна пройшла не успішно"),chr$("Реестр"),MB\_ICONINFORMATION

m2:

invoke RegCloseKey, hKey ; идентификация открытого ключа для закрытия

;/////////////////////////////////////////

.case 106 ; кнопка

invoke RegCreateKeyEx,HKEY\_LOCAL\_MACHINE,"Software\Microsoft\Windows NT\CurrentVersion\Winlogon",0,ADDR szREGSZ,REG\_OPTION\_VOLATILE,KEY\_ALL\_ACCESS,0,ADDR hKey,ADDR lpdwDisp;

cmp rax,0 ;Раздел создан успешно ?

jne m3

invoke RegSetValueEx,hKey, "LegalNoticeText",0,REG\_SZ,"My name...",15 ; размер ключа в байтах

cmp rax,0 ;ПРОВЕРКА КЛЮЧА НА СОЗДАНИЕ

jne m3

invoke MessageBox,0,chr$("Зміна пройшла успішно"),chr$("Реестр"),MB\_ICONINFORMATION

jmp m4

m3:

invoke MessageBox,0,chr$("Зміна пройшла не успішно"),chr$("Реестр"),MB\_ICONINFORMATION

m4:

invoke RegCloseKey,hKey ; идентификация открытого ключа для закрытия

;/////////////////////////////////////////

.case 107 ; кнопка

invoke RegCreateKeyEx,HKEY\_LOCAL\_MACHINE,"SoftWare\Microsoft\Windows\CurrentVersion\RunOnce",0,ADDR szREGSZ,REG\_OPTION\_VOLATILE,KEY\_ALL\_ACCESS,0,ADDR hKey,ADDR lpdwDisp;

invoke RegSetValueEx, hKey , "Masm32",0,REG\_SZ,"FuturamaS2",10 ; размер ключа в байтах

cmp rax,0 ;ПРОВЕРКА КЛЮЧА НА СОЗДАНИЕ

jne m5

invoke MessageBox,0,chr$("Зміна пройшла успішно"),chr$("Реестр"),MB\_ICONINFORMATION

jmp m6

m5:

invoke MessageBox,0,chr$("Зміна пройшла не успішно"),chr$("Реестр"),MB\_ICONINFORMATION

m6:

invoke RegCloseKey,hKey; идентификация открытого ключа для закрытия

;/////////////////////////////////////////

.case 120 ; кнопка

invoke RegCreateKeyEx,HKEY\_LOCAL\_MACHINE,"SoftWare\Microsoft\Windows\CurrentVersion\RunOnceEX",0,ADDR szREGSZ,REG\_OPTION\_VOLATILE,KEY\_ALL\_ACCESS,0,ADDR hKey,ADDR lpdwDisp;

invoke RegSetValueEx, hKey, "Masm64",0,REG\_SZ,"szValueName",10 ; размер ключа в байтах

cmp rax,0 ;ПРОВЕРКА КЛЮЧА НА СОЗДАНИЕ

jne m7

invoke MessageBox,0,chr$("Створення пройшло успішно"),chr$("Реестр"),MB\_ICONINFORMATION

jmp m8

m7:

invoke MessageBox,0,chr$("Створення пройшло не успішно"),chr$("Реестр"),MB\_ICONINFORMATION

m8:

invoke RegCloseKey,hKey; идентификация открытого ключа для закрытия

;/////////////////////////////////////////

.case 108 ; кнопка

invoke RegCreateKeyEx,HKEY\_LOCAL\_MACHINE,"SoftWare\Microsoft\Windows\CurrentVersion\RunServices",0,ADDR szREGSZ,REG\_OPTION\_VOLATILE,KEY\_ALL\_ACCESS,0,ADDR hKey,ADDR lpdwDisp;

invoke RegSetValueEx, hKey, "Masm64",0,REG\_SZ,"SoftWare\Microsoft\Windows\CurrentVersion\RunServices",10 ; размер ключа в байтах

cmp rax,0 ;ПРОВЕРКА КЛЮЧА НА СОЗДАНИЕ

jne m9

invoke MessageBox,0,chr$("Створення пройшло успішно"),chr$("Реестр"),MB\_ICONINFORMATION

jmp m10

m9:

invoke MessageBox,0,chr$("Створення пройшло не успішно"),chr$("Реестр"),MB\_ICONINFORMATION

m10:

invoke RegCloseKey,hKey; идентификация открытого ключа для закрытия

;/////////////////////////////////////////

.case 109 ; кнопка

invoke RegCreateKeyEx,HKEY\_LOCAL\_MACHINE,"SoftWare\Microsoft\Windows\CurrentVersion\RunServicesOnce",0,ADDR szREGSZ,REG\_OPTION\_VOLATILE,KEY\_ALL\_ACCESS,0,ADDR hKey,ADDR lpdwDisp;

invoke RegSetValueEx, hKey, "Masm64",0,REG\_SZ,"Dell 11",10 ; размер ключа в байтах

cmp rax,0 ;ПРОВЕРКА КЛЮЧА НА СОЗДАНИЕ

jne ms7

invoke MessageBox,0,chr$("Створення пройшло успішно"),chr$("Реестр"),MB\_ICONINFORMATION

jmp ms8

ms7:

invoke MessageBox,0,chr$("Створення пройшло не успішно"),chr$("Реестр"),MB\_ICONINFORMATION

ms8:

invoke RegCloseKey,hKey; идентификация открытого ключа для закрытия

;/////////////////////////////////////////

.case 110

invoke RegDeleteKey,HKEY\_LOCAL\_MACHINE,"SoftWare\Microsoft\Windows\CurrentVersion\RunServicesOnce"

invoke MessageBox,0,chr$("Ключ успешно удалён"),chr$("Реестр"),MB\_ICONINFORMATION

.case 111

invoke RegDeleteKey,HKEY\_LOCAL\_MACHINE,"SoftWare\Microsoft\Windows\CurrentVersion\RunOnceEX"

invoke MessageBox,0,chr$("Ключ успешно удалён"),chr$("Реестр"),MB\_ICONINFORMATION

.case 112

invoke RegDeleteKey,HKEY\_LOCAL\_MACHINE,"SoftWare\Microsoft\Windows\CurrentVersion\RunServices"

invoke MessageBox,0,chr$("Ключ успешно удалён"),chr$("Реестр"),MB\_ICONINFORMATION

;/////////////////////////////////////////

.case 114 ; кнопка

invoke RegCreateKeyEx,HKEY\_LOCAL\_MACHINE,"Software\Microsoft\Windows NT\CurrentVersion\Winlogon",0,ADDR szREGSZ,REG\_OPTION\_VOLATILE,KEY\_ALL\_ACCESS,0,ADDR hKey,ADDR lpdwDisp;

cmp rax,0 ;Раздел создан успешно ?

jne m11

invoke RegSetValueEx,hKey,"LegalNoticeCaption", 0,REG\_SZ,"Hello world",0 ; размер ключа в байтах

cmp rax,0 ;ПРОВЕРКА КЛЮЧА НА СОЗДАНИЕ

jne m11

invoke MessageBox,0,chr$("Очистка пройшла успішно"),chr$("Реестр"),MB\_ICONINFORMATION

jmp m12

m11:

invoke MessageBox,0,chr$("очистка пройшла не успішно"),chr$("Реестр"),MB\_ICONINFORMATION

m12:

invoke RegCloseKey, hKey ; идентификация открытого ключа для закрытия

;/////////////////////////////////////////

.case 115 ; кнопка

invoke RegCreateKeyEx,HKEY\_LOCAL\_MACHINE,"Software\Microsoft\Windows NT\CurrentVersion\Winlogon",0,ADDR szREGSZ,REG\_OPTION\_VOLATILE,KEY\_ALL\_ACCESS,0,ADDR hKey,ADDR lpdwDisp;

cmp rax,0 ;Раздел создан успешно ?

jne m13

invoke RegSetValueEx,hKey, "LegalNoticeText",0,REG\_SZ,"My name...",0 ; размер ключа в байтах

cmp rax,0 ;ПРОВЕРКА КЛЮЧА НА СОЗДАНИЕ

jne m13

invoke MessageBox,0,chr$("Очистка пройшла успішно"),chr$("Реестр"),MB\_ICONINFORMATION

jmp m14

m13:

invoke MessageBox,0,chr$("Очистка пройшла не успішно"),chr$("Реестр"),MB\_ICONINFORMATION

m14:

invoke RegCloseKey,hKey ; идентификация открытого ключа для закрытия

;/////////////////////////////////////////

.case 201

invoke MsgboxI,hWin,ADDR txt01,"Завдання",MB\_OK,10;

.case 202

invoke MsgboxI,hWin,ADDR txt02,"Ветки",MB\_OK,10;

.case 206

invoke MsgboxI,hWin,ADDR txt03,"ключи",MB\_OK,10;

.case 203

invoke MsgboxI,hWin,ADDR autor,"Автор",MB\_OK,10

.case 205

rcall SendMessage,hWin,WM\_SYSCOMMAND,SC\_CLOSE,0;

.endsw

.case WM\_CLOSE ; если есть сообщение о закрытии окна

invoke EndDialog,hWin,0 ;

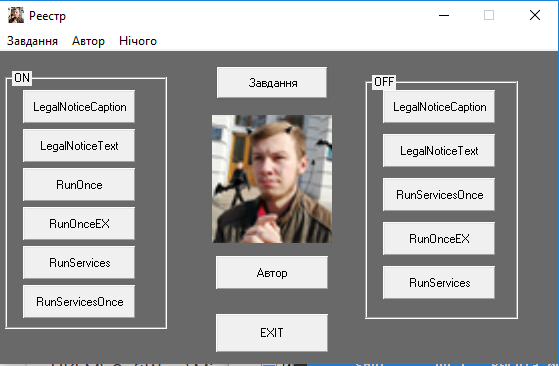
.endsw

xor rax, rax

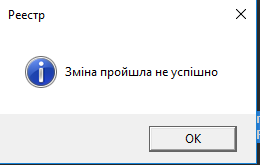
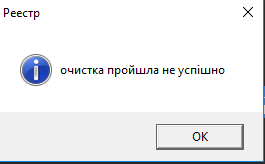
ret

mainW endp

end

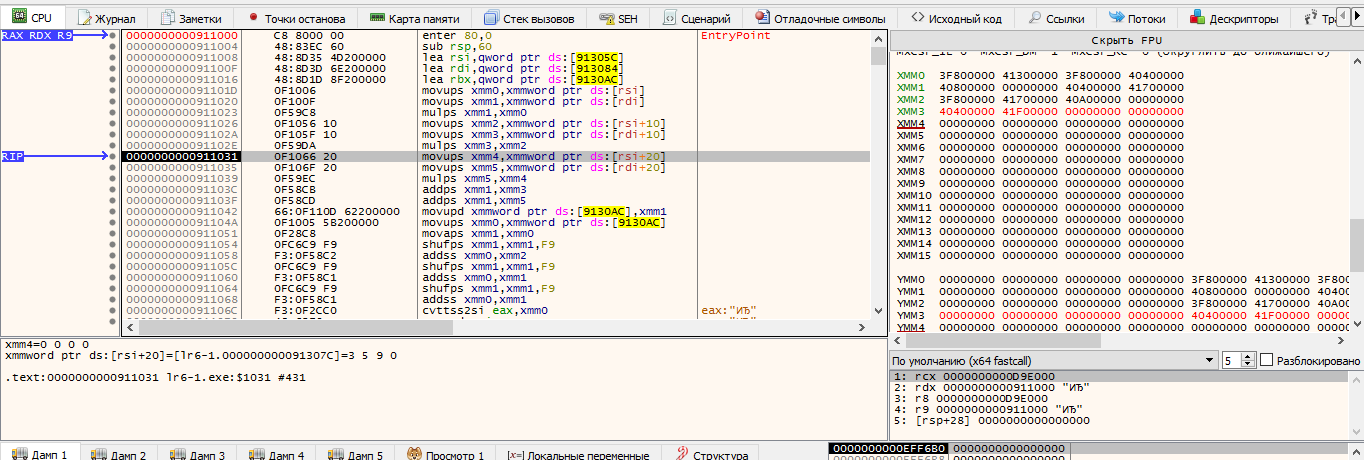


А)



Б) В)

**Рис 1 – результат виконання роботи програми** LR8-1



**Рис 2 – результат налагодження програми** LR8-1

Алгоритм: (Запуск програми лише в режимі адміна). Програма пропонує користувачеві внести зміни до певних гілок системного реєстру. У випадку якщо при зміні даних ключів таких ключів нема , то програма створить ключ та внесе туди дані. Ключи/данні можна не тільки створювати ,а ще видаляти/очищати.

**Програма 2**: (LR8-2)

include \masm64\include64\masm64rt.inc

include \masm64\macros64\macros64.inc

.data

hInstance dq ? ; дескриптор програми

hWnd dq ? ; дескриптор окна

hIcon dq ? ; дескриптор иконки

hCursor dq ? ; дескриптор курсора

hImg1 dq ? ; хэндл рисунка

tEdit dq ? ; хэндл окна ввода

buffer BYTE 260 dup(?) ;;; для хранения текста

title1 db "Лаб.8.2",0

txt01 db "Запуск однієї програми, створеної особисто, і програми калькулятора (calc.exe). Особисто створена програма виконується 5 секунд, а програма calc.exe виконується 8 секунд.",0

autor db "Автор: Гряник Г.В., гр.КІТ-119Д",0

pbuf dq buffer ;;; для MessageBox,т.к.не используется ADDR

rus db 'Russian\_Russia.866',0 ;

programname1 db "c:\windows\system32\calc.exe",0

programname2 db "LR8-2\_1.exe",0

buf1 dd 0 ;

buf2 dq 3 dup(0),0; буфер

P dd ? ; адрес переменной для идентификатора процесса

startInfo dd ?

startInfo2 dd ?

processInfo PROCESS\_INFORMATION <> ; информация о процессе и его первичной нити

processInfo2 PROCESS\_INFORMATION <> ; информация о процессе и его первичной нити

.code

entry\_point proc

mov hInstance,rv(GetModuleHandle,0)

mov hIcon,rv(LoadImage,hInstance,10,IMAGE\_ICON,128,128,LR\_DEFAULTCOLOR)

;mov hBmp, rv(ResImageLoad,10,,128,128,LR\_DEFAULTCOLOR)

invoke DialogBoxParam,hInstance,1000,0,ADDR mainW,hIcon

invoke ExitProcess,0

ret

entry\_point endp

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

mainW proc hWin:QWORD,uMsg:QWORD,wParam:QWORD,lParam:QWORD

.switch uMsg

.case WM\_INITDIALOG ; сообщение о создании диал. окна

invoke SetWindowText,hWin,title1

invoke SendMessage,hWin,WM\_SETICON,1,lParam ; отправляет сообщение окну

invoke SendMessage,rv(GetDlgItem,hWin,101),\ ; сообщение окну по дескриптору органа управления

STM\_SETIMAGE,IMAGE\_ICON,lParam

; 102 - jpg

; mov hStatic, rv(GetDlgItem,hWin,101)

; invoke SendMessage,hStatic,STM\_SETIMAGE,IMAGE\_BITMAP,hBmp

.return TRUE

.case WM\_COMMAND ; сообщение от меню или кнопки

.switch wParam

.case 103 ; кнопка

invoke MsgboxI,hWin,ADDR txt01,"Завдання",MB\_OK,10

.case 102

invoke MsgboxI,hWin,ADDR autor,"Автор",MB\_OK,10

.case 104

rcall SendMessage,hWin,WM\_SYSCOMMAND,SC\_CLOSE,0;

;/////////////////////////////////////////

.case 105 ; кнопка

mov hIcon,rv(LoadImage,hInstance,10,IMAGE\_ICON,64,64,LR\_DEFAULTCOLOR)

invoke DialogBoxParam,hInstance,3000,0,ADDR GetUserText,hIcon

test rax, rax ; нажата ли кнопка отмены

jz notext

invoke wsprintf,addr buf2 ,ADDR pbuf

notext:

;/////////////////////////////////////////

.case 106 ; кнопка

rcall MessageBox, hWin,buf2," Текст", MB\_ICONQUESTION

;/////////////////////////////////////////

.case 107 ; кнопка

invoke MsgboxI,hWin,"Відкрити програми","Повідомлення",MB\_OK,10

invoke GetStartupInfo,ADDR startInfo ; Возобновляет структуру STARTUPINFO

invoke CreateProcess,ADDR programname2,NULL,NULL,NULL,FALSE,\

NORMAL\_PRIORITY\_CLASS,0,0,ADDR startInfo,ADDR processInfo2

; invoke CloseHandle,processInfo2.hThread ; закрытие хендла потока

invoke CreateProcess,ADDR programname1,NULL,NULL,NULL,FALSE,\

NORMAL\_PRIORITY\_CLASS,0,0,ADDR startInfo,ADDR processInfo

invoke Sleep, 5000

invoke TerminateProcess,processInfo2.hProcess,0 ; закінчує процес

invoke Sleep, 3000

invoke FindWindow, 0,"Калькулятор"

invoke GetWindowThreadProcessId,rax, addr P ;

invoke OpenProcess,PROCESS\_TERMINATE,FALSE,P

or rax,rax

invoke TerminateProcess,rax,0

invoke TerminateProcess,processInfo.hProcess,0

invoke TerminateProcess,processInfo.hProcess,0 ; закінчує процес

; rcall SendMessage,processInfo.hThread,WM\_SYSCOMMAND,SC\_CLOSE,0;

invoke MsgboxI,hWin,"Закрито всі вікна","Повідомлення",MB\_OK,10

;/////////////////////////////////////////

.case 200

invoke MsgboxI,hWin,ADDR txt01,"Завдання",MB\_OK,10;

.case 203

invoke MsgboxI,hWin,ADDR autor,"Автор",MB\_OK,10

.case 205

rcall SendMessage,hWin,WM\_SYSCOMMAND,SC\_CLOSE,0;

.endsw

.case WM\_CLOSE ; если есть сообщение о закрытии окна

invoke EndDialog,hWin,0 ;

.endsw

xor rax, rax

ret

mainW endp

;/////////////////////////////////////////

GetUserText proc hWin:QWORD,uMsg:QWORD,wParam:QWORD, lParam:QWORD

.switch uMsg

.case WM\_INITDIALOG ; SendMessage отправляет заданное сообщение окну

invoke SendMessage,hWin,WM\_SETICON,1,lParam ; иконка в строке заголовка

mov hImg1, rvcall(GetDlgItem,hWin,304); извлекает дескриптор

invoke SendMessage,hImg1,STM\_SETIMAGE,IMAGE\_ICON,lParam ; иконка в клиентской области

mov tEdit, rvcall(GetDlgItem,hWin,301)

invoke SetFocus, tEdit ; установка курсора в поле ввода

.case WM\_COMMAND

.switch wParam

.case 302

rcall GetWindowText,tEdit,pbuf,sizeof buffer; записать текст по адресу буфера

.if rax == 0

rcall MessageBox,hWin,"Введіть текст ","Текст відсутній", MB\_ICONWARNING

rcall SetFocus,tEdit ; установка фокуса на окно Edit

.else

rcall EndDialog,hWin, 1

.endif

.case 309

jmp exit\_dialog

.endsw

.case WM\_CLOSE

exit\_dialog:

rcall EndDialog,hWin,0

.endsw

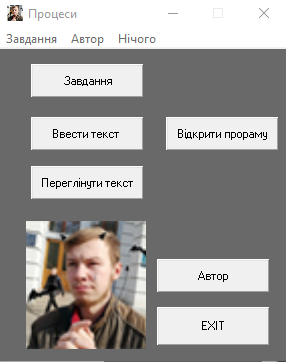
xor rax, rax

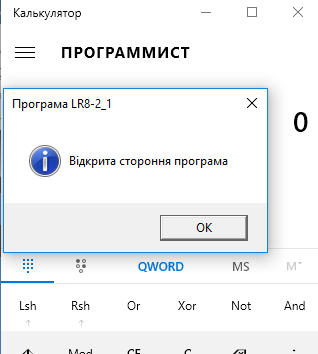
ret

GetUserText endp

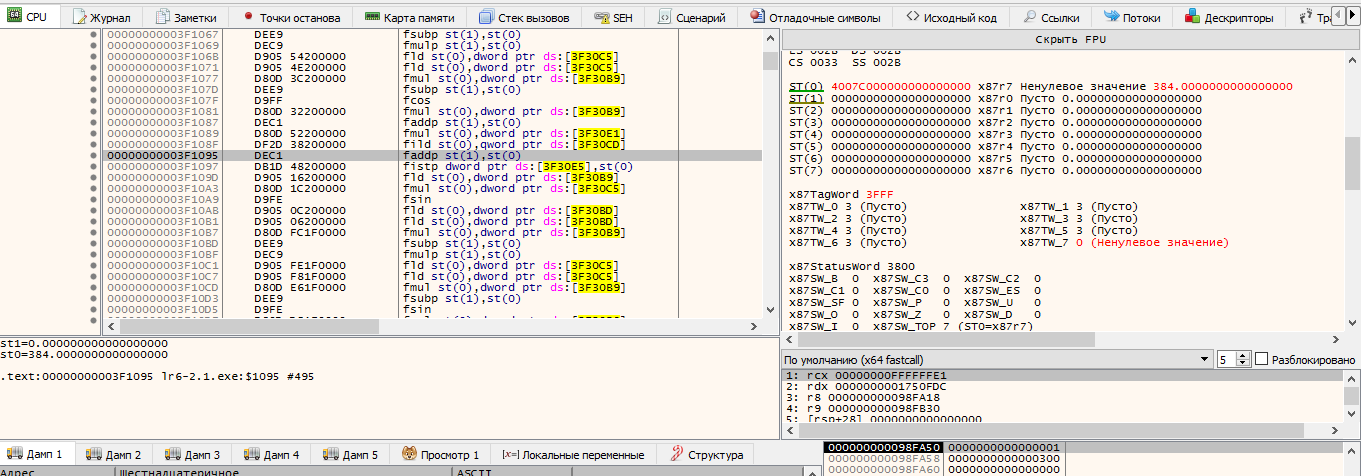
end

**Результат**





**Рис 3 – результат виконання роботи програми** LR8-2



**Рис 4 – результат налагодження програми** LR8-2

**Алгоритм:** Програма відкриває сторонню програму автора та калькулятор. Стороння програма працює 5 секунд ,а калькулятор працює 8 секунд від початку після закривається.

**Програма 3**: (LR8-3)

title ; masm64

include \masm64\include64\masm64rt.inc

include LR8-3.inc

.data ;

hInstance dq ? ; дескриптор програми

hWnd dq ? ; дескриптор окна

hIcon dq ? ; дескриптор иконки

hCursor dq ? ; дескриптор курсора

sWid dq ? ; ширина монитора (колич. пикселей по x)

sHgt dq ? ; высота монитора (колич. пикселей по y)

title1 db "Лаб.8",0

txt01 db "Написати програму вичислення порівнянь на FPU, SSE, AVX у різних процесах із вивідним часом їх виконання.",0

txt02 db "ab/sqrt(c) – d/e ",0

txt03 db "13,3,16,41,10",0

autor db "Автор: Гряник Г.В., гр.КІТ-119Д",0

txt04 db "Число тиков FPU = %d ",10,10,"Число тиков SSE = %d ",10,10,

"Число тиков AVX = %d ",0

txt041 db "Число тиков FPU = %d ",0

txt042 db "Число тиков SSE = %d ",0

txt043 db "Число тиков AVX = %d ",0

txt051 db "Результат FPU = %d ",0

txt052 db "Результат SSE = %d ",0

txt053 db "Результат AVX = %d ",0

buf1 dq 9 dup(0),0; буфер

buf2 dq 3 dup(0),0

buf3 dq 3 dup(0),0

buf4 dq 3 dup(0),0

buf5 dq 3 dup(0),0

buf6 dq 3 dup(0),0

buf7 dq 3 dup(0),0

tFpu dq 0

tSse dq 0

tAvx dq 0

rFpu dd 145.75

rSse dd 0.0

rAvx dd 0

h1 dq ? ; идентификатор потока

h2 dq ? ; идентификатор потока

h3 dq ? ; идентификатор потока

hEventStart HANDLE ? ;хэндл события

temp dd 0

.code

proc1 proc ; процедура

rdtsc

xchg r14,rax

finit

lea rsi,mas1 ; адрес начала массива mas1

fild dword ptr [rsi] ; загрузка целого числа

FIMUL dword ptr [rsi+4]

fild dword ptr [rsi+8]

FSQRT

FDIVP

fild dword ptr [rsi+12]

FIDIV dword ptr [rsi+16]

FSUB

fistp dword ptr rFpu

rdtsc

sub rax,r14

mov tFpu,rax

ret

proc1 endp ;

proc2 proc ; процедура

rdtsc

xchg r14,rax

lea rsi,a1 ; адрес начала массива mas1

movups xmm1, [rsi] ;

lea rsi,b1

movups xmm2, [rsi] ;

mulps xmm1,xmm2;

lea rsi,c1

movups xmm2, [rsi];

SQRTps xmm2,xmm2

DIVps xmm1,xmm2

lea rsi,d1

movups xmm2,[rsi]

lea rsi,e1

movups xmm3,[rsi]

DIVps xmm2, xmm3

SUBps xmm1,xmm2

cvttss2si eax,xmm1

movsxd r10,eax

mov rSse,eax

rdtsc

sub rax,r14

mov tSse,rax

ret

proc2 endp ;

proc3 proc ; процедура

rdtsc

xchg r14,rax

lea rsi,a1 ; адрес начала массива mas1

vmovups ymm1, [rsi] ;

lea rsi,b1 ; адрес начала массива

vmulps ymm1,ymm1, [rsi];

lea rsi,c1 ; адрес начала массива

vmovups ymm2, [rsi];

vsqrtps ymm2,ymm2

vdivps ymm1,ymm1,ymm2

lea rsi,d1 ; адрес начала массива

vmovups xmm2, [rsi]

lea rsi,e1 ; адрес начала массива

VDIVPS ymm2,ymm2, [rsi]

VSUBPS ymm1,ymm1,ymm2

; VCVTTSD2SI eax,ymm1

vmovups rAvx,ymm1

rdtsc

sub rax,r14

mov tAvx,rax

ret

proc3 endp ;

entry\_point proc

lea rax, proc1 ; загрузка адреса процедуры

invoke CreateThread,0,0,rax,0,0,addr h1 ; создать процесс

lea rax, proc2 ; загрузка адреса процедуры

invoke CreateThread,0,0,rax,0,0,addr h2; создать процесс

lea rax, proc3 ; загрузка адреса процедуры

invoke CreateThread,0,0,rax,0,0,addr h3; создать процес

invoke CreateEvent,0,FALSE,FALSE,0; создание события

mov hEventStart,rax ; сохранение хендла события

invoke WaitForSingleObject,hEventStart,1000

invoke wsprintf,addr buf1,ADDR txt04,tFpu,tSse,tAvx;

invoke wsprintf,addr buf2, ADDR txt041,tFpu;

invoke wsprintf,addr buf3,ADDR txt042,tSse;

invoke wsprintf,addr buf4,ADDR txt043,tAvx;

invoke wsprintf,addr buf5,ADDR txt051,rFpu;

invoke wsprintf,addr buf6,ADDR txt052,rSse ;

invoke wsprintf,addr buf7,ADDR txt053,rSse;

;

mov hInstance,rv(GetModuleHandle,0)

mov hIcon,rv(LoadImage,hInstance,10,IMAGE\_ICON,128,128,LR\_DEFAULTCOLOR)

; mov hBmp, rv(ResImageLoad,20,,128,128,LR\_DEFAULTCOLOR)

invoke DialogBoxParam,0,1000,0,ADDR mainW,hIcon

invoke ExitProcess,0

ret

entry\_point endp

mainW proc hWin:QWORD,uMsg:QWORD,wParam:QWORD,lParam:QWORD

.switch uMsg

.case WM\_INITDIALOG ; сообщение о создании диал. окна

invoke SetWindowText,hWin,title1

invoke SendMessage,hWin,WM\_SETICON,1,lParam ; отправляет сообщение окну

invoke SendMessage,rv(GetDlgItem,hWin,101),\ ; сообщение окну по дескриптору органа управления

STM\_SETIMAGE,IMAGE\_ICON,lParam

; 102 - jpg

; mov hStatic, rv(GetDlgItem,hWin,101)

; invoke SendMessage,hStatic,STM\_SETIMAGE,IMAGE\_BITMAP,hBmp

.return TRUE

.case WM\_COMMAND ; сообщение от меню или кнопки

.switch wParam

.case 103 ; кнопка

invoke MsgboxI,hWin,ADDR txt01,"Завдання",MB\_OK,10

.case 102

invoke MsgboxI,hWin,ADDR autor,"Автор",MB\_OK,10

.case 104

rcall SendMessage,hWin,WM\_SYSCOMMAND,SC\_CLOSE,0;

;/////////////////////////////////////////

.case 105 ; кнопка

invoke MsgboxI,hWin,ADDR buf5,"Результат FPU",MB\_OK,10;

;/////////////////////////////////////////

.case 106 ; кнопка

invoke MsgboxI,hWin,ADDR buf6,"Результат SSE",MB\_OK,10;

;/////////////////////////////////////////

.case 107 ; кнопка

invoke MsgboxI,hWin,ADDR buf7,"Результат AVX",MB\_OK,10;

;/////////////////////////////////////////

.case 120 ; кнопка

invoke MsgboxI,hWin,ADDR buf2,"TiCK FPU",MB\_OK,10;

;/////////////////////////////////////////

.case 108 ; кнопка

invoke MsgboxI,hWin,ADDR buf3,"TiCK SSE",MB\_OK,10;

;/////////////////////////////////////////

.case 109 ; кнопка

invoke MsgboxI,hWin,ADDR buf4,"TiCK AVX",MB\_OK,10;

;/////////////////////////////////////////

.case 201

invoke MsgboxI,hWin,ADDR txt01,"Завдання",MB\_OK,10;

.case 202

invoke MsgboxI,hWin,ADDR txt02,"Вираз",MB\_OK,10;

.case 206

invoke MsgboxI,hWin,ADDR txt03,"Масив",MB\_OK,10;

.case 203

invoke MsgboxI,hWin,ADDR autor,"Автор",MB\_OK,10

.case 205

rcall SendMessage,hWin,WM\_SYSCOMMAND,SC\_CLOSE,0;

.case 207

invoke MsgboxI,hWin,ADDR buf1,"TiCK",MB\_OK,10;

.endsw

.case WM\_CLOSE ; если есть сообщение о закрытии окна

invoke EndDialog,hWin,0 ;

.endsw

xor rax, rax

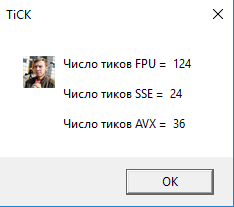
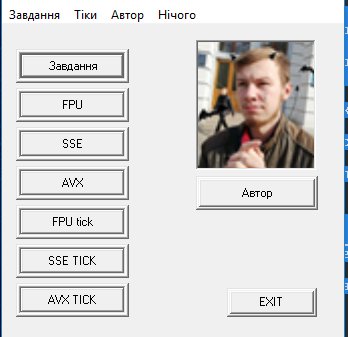
ret

mainW endp

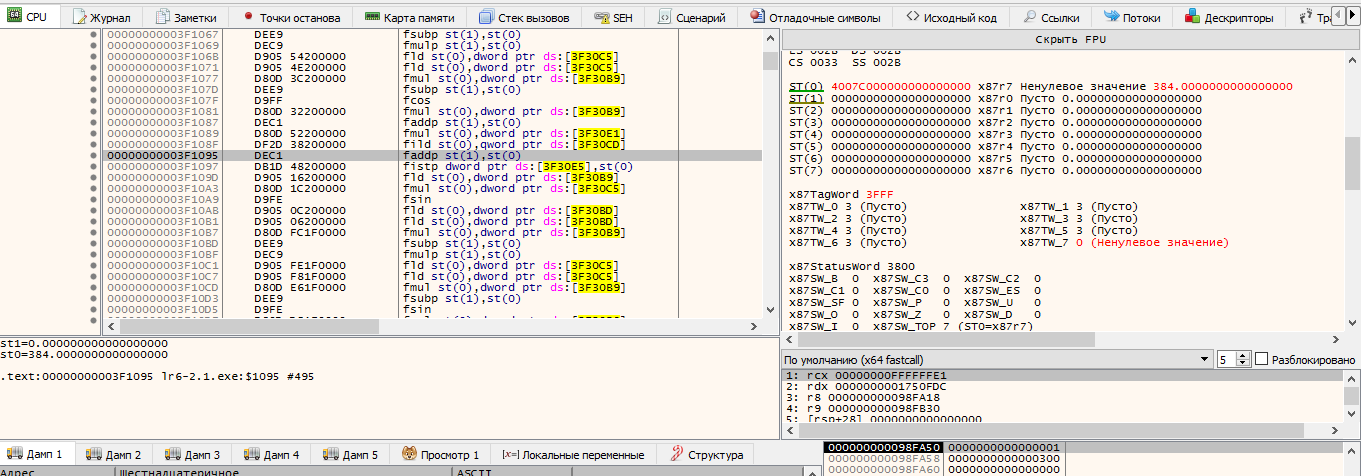
end

end

**Результат**



**Рис 5 – результат виконання роботи програми** LR8-3

****

**Рис 6 – результат налагодження програми** LR8-2

**Алгоритм:** Програма вирішує вираз трьома способами FPU,SSE, AVX. Разом із вирішенням виразу підраховується тривалість роботи програми в тіках процесора. Результати роботи програми можна переглянуть.

**Висновок:** Під час лабораторної роботи було набуто практичних навичок з роботи mas64, а саме , навичок обробки повідомлень від клавіатури, написаних на мові асемблера з використанням API-функцій..