Міністерство освіти та науки України

**Кіровоградський Державний Технічний університет**

Кафедра програмного забезпечення

Курсова робота

з дисципліни “Програмування на мові ASM-86” на тему:

**Текстовий редактор під Windows**

Зміст

1. Вступ

2. Постановка задачі

3. Обґрунтування вибору методів розв’язку задачі

4. Алгоритм програми

5. Реалізація програми

6. Системні вимоги

7. Інструкція для користувача

8. Висновки

9. Використана література

Додаток

## 1. Вступ

Тепер комп’ютери відіграють у житті людини все більшу та більш роль. Раніше, коли ще не було комп’ютерів, чи вони були мало розповсюдженні, все робилося вручну. Коли комп’ютери одержали широке розповсюдження, комп’ютер став допомагати людині, бо він може багато операцій робити набагато швидше, ніж людина.

Інформація - цінна річ, тому для її збереження в сучасному суспільстві використовують все частіше персональний комп‘ютер. При перегляді текстових файлів виникає потреба в зручних в користуванні та розумінні програм переглядачів текстових файлів.

## 2. Постановка задачі

Розробити текстовий редактор під Windows (типу Notepad) на мові програмування ASM-86. Програма повинна виконувати основні дії над введеним текстом.

## 3. Обґрунтування вибору методів розв’язку задачі

Отже, необхідно написати текстовий редактор під Windows. Дана програма призначена для перегляду текстових файлів різного розміру. Програма може бути використана будь яким користувачем ЕОМ для перегляду потрібної інформації. При виборі алгоритму програми у мене не виникло варіантів. Тут все дуже просто. В Windows вже є функції API для роботи з текстом, іх тільки потрібно правильно викликати. Ще потрібно правильно написати обробники повідомлень.

## 4. Алгоритм програми

Читаємо командний рядок.

Читаємо іконку та курсор з ресурсів.

Будування головного вікна.

Ініціалізація акселераторів.

Отримання повідомлення

Якщо це WM\_QUIT, то перехід до пункту 11.

Перетворимо акселератори в IDM\*

Перетворимо повідомлення від клавіш

Відішлемо назад.

Перехід до пункту 5.

Закінчити програму.

## 5. Реалізація програми

Програма написана на мові ASM-86 з використанням команд процесора 386. Вона складається з головної програми, яка читає командний рядок, будує головне вікно, та організує головний цикл програми. Та з обробників повідомлень вікна, які і викликають функції для роботи з текстом.

Вона працює на будь якій машині де WINDOWS. При написанні програми для реалізації роботи з текстом та файлом були використані функції API WINDOWS.

Програма компілюється Turbo Assembler, зв’язується за допомогою TLINK32, та компілюються ресурси за допомогою brcc32.

tasm /m /ml /D\_TASM\_ winpad. asm

brcc32 winpad. rc

tlink32 /Tpe /aa /c /x winpad. obj,,,,,winpad. res

## 6. Системні вимоги

IBM сумісний комп’ютер із мікропроцесором Intel 80386 або старшим.

Операційна система - Windows 3.1 або старша

Пам’ять - 2M і вище

## 7. Інструкція для користувача

Для запуску текстового редактора під Windows, потрібно зайти в меню “Пуск", вибрати “Выполнить” та в командному рядку набрати Notepad. Exe та натиснути Enter. Також можна набрати ім’я файлу, який потрібно редагувати.

Після запуску програми з’являється оболонка, в якій вже можна набирати текст, причому, можна використовувати операції з блоком. Також, в програмі є меню, в якому є опції для читання файлу, для запису файлу, для операцій з буфером, та вивід інформації про програму.

## 8. Висновки

Отже, на мові ASM-86 створено текстовий редактор під Windows (типу Notepad). При постійній роботі з комп‘ютером досить часто потрібно переглянути файл. Тому дана програма є незамінною в роботі. Вона є простою в користуванні та розумінні. Під час написання цієї програми був розібраний метод писання програм під Windows.

## 9. Використана література

1) “Програмування на мові Асемблера ЕС ЕВМ" З.С. Брич, В.I. Воющ Москва 1989 р. ст. 166

2).“Програмування на сучасних мовах програмування”. *Москва 1990 р. ст. 206*

3) “Основи мови” Асемблер” В.I. Криволап. *Москва 1997р. ст.309*

4) “Ассемблер для початкiвцiв” М.П. Шукiн. Київ 1980р. ст.155

5) “Турбо Асемблер” Л.В. Захаров. Харків 1995 р ст.178

6) “Макро Асемблер” К.С. Кропiйко, О.Д. Богатирова. Київ - “Наука” 1991р.

7) Електронний довідник BOOK.

## Додаток

## Лістинг програми

**; winpad. asm**

; Графическое win32-приложение - текстовый редактор

;

; Компиляция TASM

; tasm /m /ml /D\_TASM\_ winpad. asm

; brcc32 winpad. rc

; tlink32 /Tpe /aa /c /x winpad. obj,,,,,winpad. res

include def32. inc

include user32. inc

include kernel32. inc

include comdlg32. inc

ID\_MENUequ700h

ID\_ACCELequ701h

ID\_ABOUTequ702h

MAXSIZE equ 260; максимальное имя файла

MEMSIZEequ 65535; максимальный размер временного буфера в памяти

EditIDequ 1

.386

. model flat

. const

c\_w\_namedb 'Asmpad95',0; это и имя класса и имя основного окна

edit\_classdb 'edit',0; предопределённое имя класса для редактора

changes\_msgdb 'Save changes? ',0

filter\_stringdb 'All Files',0,'\*. \*',0; маски для Get\*FileName

db 'Text Files',0,'\*. txt',0,0

. data

; структура, использующаяся Get\*FileName

ofnOPENFILENAME <SIZE ofn,?,?,offset filter\_string,?,?,?,offset buffer,MAXSIZE,0,?,?,?,?,?,?,0,?,?,? >

; стурктура, описывающая наш основной класс

wcWNDCLASSEX <SIZE WNDCLASSEX,CS\_HREDRAW or CS\_VREDRAW,offset win\_proc,0,0,?,?,?,COLOR\_WINDOW+1, ID\_MENU,offset c\_w\_name,0>

flag\_untitleddb1; = 1 если имя файла не определено (новый файл)

. data?

h\_editwindowdd? ; идентификатор окна редактора

h\_acceldd? ; идентификатор массива акселераторов

p\_memorydd? ; адрес буфера в памяти

SizeReadWritedd?

msg\_MSG<>

recRECT<>

bufferdbMAXSIZE dup (?) ; имя файла

window\_titledbMAXSIZE dup (?), 12 dup (?)

. code

\_start:

callGetCommandLine; получим нашу командную строку

movedi,eax

moval,' '

movecx,MAXSIZE

repnescasb; найдём конец имени нашей программы

cmpbyte ptr [edi],0

jecmdline\_empty

movesi,edi

movedi,offset buffer

repmovsb

movflag\_untitled,0

cmdline\_empty:

; подготовим и зарегистрируем класс

xorebx,ebx

callGetModuleHandle; определим наш идентификатор

movesi,eax

movwc. hInstance,eax; и сохраним его в wc. hInstance

movofn. \_hInstance,eax

pushIDI\_APPLICATION; или IDI\_ICON если иконка есть в ресурсах

pushebx; или esi если иконка есть в ресурсах

callLoadIcon

movwc. hIcon,eax

pushIDC\_ARROW; предопределённый курсор (стрелка)

pushebx

callLoadCursor

movwc. hCursor,eax

pushoffset wc

callRegisterClassEx

; создадим основное окно

pushebx

pushesi

pushebx

pushebx

push400

push600

pushCW\_USEDEFAULT

pushCW\_USEDEFAULT

pushWS\_OVERLAPPEDWINDOW

pushoffset c\_w\_name

pushoffset c\_w\_name

pushWS\_EX\_CLIENTEDGE

callCreateWindowEx

pusheax; для pop esi перед message\_loop

pusheax

pushSW\_SHOWNORMAL

pusheax

callShowWindow

callUpdateWindow

; инициализируем акселераторы

pushID\_ACCEL

pushesi

callLoadAccelerators

movh\_accel,eax

; цикл ожидания сообщения

popesi; ESI - идентификатор основного окна

movedi,offset msg\_; EDI - структура с сообщением от него

message\_loop:

pushebx

pushebx

pushebx

pushedi

callGetMessage; получим сообщение

testeax,eax; если это WM\_QUIT

jzexit\_msg\_loop; выйти из цикла

pushedi

pushh\_accel

pushesi; hWnd

callTranslateAccelerator; преобразуем акселераторы в IDM\*

testeax,eax

jnzmessage\_loop

pushedi

callTranslateMessage; преобразуем сообщения от клавиш

pushedi

callDispatchMessage; и отошлём обратно

jmp short message\_loop

exit\_msg\_loop:

pushmsg\_. wParam

callExitProcess; конец программы

; процедура win\_proc

;

; процедура не должна изменять регистры EBP,EDI,ESI и EBX!

win\_proc proc near

; параметры (с учётом push ebp)

wp\_hWnd equ dword ptr [ebp+08h]

wp\_uMsg equ dword ptr [ebp+0Ch]

wp\_wParam equ dword ptr [ebp+10h]

wp\_lParam equ dword ptr [ebp+14h]

; инициализируем стековый кадр

pushebp

movebp,esp; создадим стековый кадр

pusha; сохраним все регистры

xorebx,ebx; 0 для команд push 0

movesi,wp\_hWnd; для команд push hWnd

moveax,wp\_uMsg

; обработаем пришедшее сообщение

cmpeax,WM\_CREATE

jeh\_wm\_create

cmpeax,WM\_SIZE

jeh\_wm\_size

cmpeax,WM\_DESTROY

jeh\_wm\_destroy

cmpeax,WM\_COMMAND

jeh\_wm\_command

cmpeax,WM\_ACTIVATE

jeh\_wm\_activate

cmpeax,WM\_CLOSE

jeh\_wm\_close

def\_proc:

popa

leave; если это ненужное сообщение

jmpDefWindowProc; оставить его обработчику по умолчанию

;

; обработчик WM\_CLOSE

; если нужно, спрашивает, сохранить ли файл

h\_wm\_close:

callsave\_contents

jmp short def\_proc

;

; обработчик WM\_CREATE

;

h\_wm\_create:

; здесь также можно создать toolbar и statusbar

; создадим окно редактора

pushebx

pushwc. hInstance; идентификатор основной программы

pushEditID

pushesi; hWnd

pushebx; 0

pushebx; 0

pushebx; 0

pushebx; 0

pushWS\_VISIBLE or WS\_CHILD or ES\_LEFT or ES\_MULTILINE or ES\_AUTOHSCROLL or ES\_AUTOVSCROLL

pushebx; 0

pushoffset edit\_class

pushebx; 0

callCreateWindowEx

movh\_editwindow,eax

; передадим ему фокус

pusheax

callSetFocus

cmpflag\_untitled,1

jecontinue\_create

callskip\_getopen; откроем файл, указанный в командной строке

continue\_create:

callset\_title

jmp end\_wm\_check

;

; обработчик WM\_COMMAND

;

h\_wm\_command:

moveax,wp\_wParam

cwde; low word contains IDM\_\*

subeax,100h

jbdef\_proc

; обработаем сообщения от пунктов меню

call dword ptr menu\_handlers [eax\*4]

jmp end\_wm\_check

menu\_handlersdd offset h\_idm\_new,offset h\_idm\_open,offset h\_idm\_save

dd offset h\_idm\_saveas,offset h\_idm\_exit,offset h\_idm\_about

dd offset h\_idm\_undo, offset h\_idm\_cut, offset h\_idm\_copy

dd offset h\_idm\_paste, offset h\_idm\_clear, offset h\_idm\_setsel

; сообщения от пунктов меню должны быть описаны в win95pad. rc именно в таком

; порядке - от IDM\_NEW 100h до IDM\_CLEAR 10Ah

h\_idm\_setsel:

push-1; - 1

pushebx; 0

pushEM\_SETSEL; выделить весь текст

pushh\_editwindow

callSendMessage

ret

; обработчики сообщений из меню EDIT:

h\_idm\_clear:

moveax,WM\_CLEAR

jmp short send\_to\_editor

h\_idm\_paste:

moveax,WM\_PASTE

jmp short send\_to\_editor

h\_idm\_copy:

moveax,WM\_COPY

jmp short send\_to\_editor

h\_idm\_cut:

moveax,WM\_CUT

jmp short send\_to\_editor

h\_idm\_undo:

moveax,EM\_UNDO

send\_to\_editor:

pushebx; 0

pushebx; 0

pusheax

pushh\_editwindow

callSendMessage

ret

; обработчик IDM\_NEW

h\_idm\_new:

callsave\_contents; запишем файл если нужно

movbyte ptr flag\_untitled,1

callset\_title; отметим что файл не назван

pushebx

pushebx

pushWM\_SETTEXT

pushh\_editwindow

callSendMessage; пошлём пустой WM\_SETTEXT редактору

ret

; обработчик IDM\_ABOUT

h\_idm\_about:

pushebx; 0

pushoffset about\_proc

pushesi; hWnd

pushID\_ABOUT

pushwc. hInstance

callDialogBoxParam

ret

; обработчик IDM\_SAVEAS и IDM\_SAVE

h\_idm\_save:

cmpflag\_untitled,1; если файл назван

jneskip\_getsave; пропустим вызов GetSaveFileName

h\_idm\_saveas:

; спросим имя файла

movofn. Flags,OFN\_EXPLORER or OFN\_OVERWRITEPROMPT

pushoffset ofn

callGetSaveFileName

testeax,eax

jzfile\_save\_failed

skip\_getsave:

; создадим его

pushebx

pushFILE\_ATTRIBUTE\_ARCHIVE

pushCREATE\_ALWAYS

pushebx

pushFILE\_SHARE\_READ or FILE\_SHARE\_WRITE

pushGENERIC\_READ or GENERIC\_WRITE

pushoffset buffer

callCreateFile

movedi,eax

; выделим память

pushMEMSIZE

pushGMEM\_MOVEABLE or GMEM\_ZEROINIT

callGlobalAlloc

pusheax; hMemory для GlobalFree

pusheax; hMemory для GlobalLock

callGlobalLock

movesi,eax; адрес буфера в ESI

; заберём текст из редактора

pushesi

pushMEMSIZE-1

pushWM\_GETTEXT

pushh\_editwindow

callSendMessage

; запишем в файл

pushesi; pMemory

calllstrlen

pushebx

pushoffset SizeReadWrite

pusheax; размер буфера

pushesi; адрес буфера

pushedi; идентификатор файла

callWriteFile

pushesi; pMemory

callGlobalUnlock

callGlobalFree; hMemory уже в стеке

pushedi; идентификатор файла

callCloseHandle

; сбросим флаг модификации в редакторе

pushebx

pushebx

pushEM\_SETMODIFY

pushh\_editwindow

callSendMessage

movbyte ptr flag\_untitled,0

callset\_title

file\_save\_failed:

pushh\_editwindow

callSetFocus

ret

; обработчик IDM\_OPEN

h\_idm\_open:

callsave\_contents

; вызовем стандартный диалог выбора имени файла

movofn. Flags, OFN\_FILEMUSTEXIST or OFN\_PATHMUSTEXIST or OFN\_EXPLORER

pushoffset ofn

callGetOpenFileName

testeax,eax

jzfile\_open\_failed

skip\_getopen:

; откроем выбранный файл

pushebx

pushFILE\_ATTRIBUTE\_ARCHIVE

pushOPEN\_EXISTING

pushebx

pushFILE\_SHARE\_READ or FILE\_SHARE\_WRITE

pushGENERIC\_READ or GENERIC\_WRITE

pushoffset buffer

callCreateFile

movedi,eax; идентификатор для ReadFile

; выделим память

pushMEMSIZE

pushGMEM\_MOVEABLE or GMEM\_ZEROINIT

callGlobalAlloc

pusheax; hMemory для GlobalFree

pusheax; hMemory для GlobalLock

callGlobalLock; получим адрес выделенной памяти

pusheax; pMemory для GlobalUnlock

pusheax; pMemory для SendMessage

; прочитаем файл

pushebx

pushoffset SizeReadWrite

pushMEMSIZE-1

pusheax; pMemory для ReadFile

pushedi

callReadFile

; пошлём окну редактора сообщение wm\_settext чтобы он забрал текст из буфера

pushebx; pMemory уже в стеке

pushWM\_SETTEXT

pushh\_editwindow

callSendMessage

; а теперь можно закрыть файл и освободить память

callGlobalUnlock; pMemory уже в стеке

callGlobalFree; hMemory уже в стеке

pushedi; hFile

callCloseHandle;

movbyte ptr flag\_untitled,0

callset\_title

file\_open\_failed:

pushh\_editwindow

callSetFocus

ret

; обработчик IDM\_EXIT

h\_idm\_exit:

callsave\_contents

pushesi; hWnd

callDestroyWindow; уничтожить наше окно

ret

;

; обработчик WM\_SIZE

;

h\_wm\_size:

; здесь также надо послать WM\_SIZE окнам toolbar и statusbar

; изменим размер окна редактора так чтобы оно по-прежнему было на всё окно

pushoffset rec

pushesi; hWnd

callGetClientRect

push1; true

pushrec. bottom; height

pushrec. right; width

pushebx; y

pushebx; x

pushh\_editwindow

callMoveWindow

jmp short end\_wm\_check

;

; обработчик WM\_ACTIVATE

;

h\_wm\_activate:

pushh\_editwindow

callSetFocus

jmp short end\_wm\_check

;

; обработчик WM\_DESTROY

;

h\_wm\_destroy:

pushebx

callPostQuitMessage; послать WM\_QUIT основной программе

end\_wm\_check:

popa

xor eax,eax; вернуть 0

leave

ret16

; процедура set\_title

; устанавливает новый заголовок для основного окна

set\_title:

pushesi

pushedi

movedi,offset window\_title

cmpbyte ptr flag\_untitled,1; если у файла нет имени

jeuntitled; использовать Untitled

movesi,ofn. lpstrFile; [ESI] - имя файла с путём

movzxeax,ofn. nFileOffset; eax - начало имени файла

addesi,eax

copy\_filename:

lodsb; скопирем файл побайтово в название окна

testal,al

jzadd\_progname; пока не встретится ноль

stosb

jmp short copy\_filename

add\_progname:

movdword ptr [edi],' - '; припишем минус

addedi,3

movesi,offset c\_w\_name

movecx,9; и название программы

rep movsb

popedi

popesi

pushoffset window\_title

pushesi; идентификатор окна

callSetWindowText

ret

untitled:

movdword ptr [edi],'itnU'; допишем "Unti"

movdword ptr [edi+4],'delt'; допишем "tled"

addedi,8

jmp short add\_progname

; процедура save\_contents

; EBX=0, ESI=hWnd

save\_contents:

; спросим редактор, изменялся ли текст

pushebx

pushebx

pushEM\_GETMODIFY

pushh\_editwindow

callSendMessage

testeax,eax

jznot\_modified

; спросим пользователя, сохранять ли его

pushMB\_YESNO + MB\_ICONWARNING

pushoffset c\_w\_name

pushoffset changes\_msg

pushesi

callMessageBox

cmpeax, IDYES

jnenot\_modified

; сохраним его

callh\_idm\_save

not\_modified:

ret

win\_proc endp

about\_proc proc near

; параметры (с учётом push ebp)

ap\_hDlg equ dword ptr [ebp+08h]

ap\_uMsg equ dword ptr [ebp+0Ch]

ap\_wParam equ dword ptr [ebp+10h]

ap\_lParam equ dword ptr [ebp+14h]

pushebp

movebp,esp; создадим стековый кадр

cmpap\_uMsg,WM\_COMMAND

jnedont\_proceed

cmpap\_wParam, IDOK

jnedont\_proceed

push1

pushap\_hDlg

callEndDialog

dont\_proceed:

xoreax,eax; not processed

leave

ret 16

about\_proc endp

end\_start

**; comdlg32. inc**

; включаемый файл с определениями функций из comdlg32. dll

;

ifdef \_TASM\_

includelib import32. lib

; имена используемых функций

extrnGetOpenFileNameA: near

extrnGetSaveFileNameA: near

; присваивания для облегчения читаемости кода

GetOpenFileNameequGetOpenFileNameA

GetSaveFileNameequGetSaveFileNameA

else

includelib comdlg32. lib

; истинные имена используемых функций

extrn\_\_imp\_\_GetOpenFileNameA@4: dword

extrn\_\_imp\_\_GetSaveFileNameA@4: dword

; присваивания для облегчения читаемости кода

GetOpenFileNameequ\_\_imp\_\_GetOpenFileNameA@4

GetSaveFileNameequ\_\_imp\_\_GetSaveFileNameA@4

Endif

**; def32. inc**

; файл с определениями констант и типов для примеров под win32

; из winbase. h

STD\_OUTPUT\_HANDLEequ-11

OPEN\_EXISTINGequ3

CREATE\_ALWAYSequ2

GMEM\_MOVEABLEequ2

GMEM\_ZEROINITequ40h

; из windef. h

RECTstruc

leftdd?

topdd?

rightdd?

bottomdd?

RECTends

; из winerror. h

NO\_ERRORequ0

ERROR\_NO\_MORE\_ITEMSequ259

; из winnetwk. h

RESOURCEUSAGE\_CONNECTABLEequ 1

RESOURCETYPE\_ANYequ0

RESOURCE\_CONNECTEDequ1

NTRESOURCE struc

dwScopedd?

dwTypedd?

dwDisplayTypedd?

dwUsagedd?

lpLocalNamedd?

lpRemoteNamedd?

lpCommentdd?

lpProviderdd?

NTRESOURCE ends

; из winnt. h

GENERIC\_READequ80000000h

GENERIC\_WRITEequ40000000h

FILE\_SHARE\_READequ1

FILE\_SHARE\_WRITEequ2

FILE\_ATTRIBUTE\_ARCHIVEequ20h

; из winuser. h

MB\_OKequ0

MB\_ICONINFORMATIONequ40h

IDC\_ARROWequ32512

IDI\_APPLICATIONequ32512

WS\_OVERLAPPEDWINDOW equ0CF0000h

WM\_DESTROYequ2

WM\_COMMANDequ111h

WM\_INITDIALOGequ110h

WM\_CLOSEequ10h

CS\_HREDRAWequ2

CS\_VREDRAWequ1

CW\_USEDEFAULTequ80000000h

SW\_SHOWNORMALequ1

COLOR\_WINDOWequ5

BN\_CLICKEDequ0

WNDCLASSEXstruc

cbSizedd?

styledd?

lpfnWndProcdd?

cbClsExtradd?

cbWndExtradd?

hInstancedd?

hIcondd?

hCursordd?

hbrBackgrounddd?

lpszMenuNamedd?

lpszClassNamedd?

hIconSmdd?

WNDCLASSEXends

MSGstruc

hwnddd?

messagedd?

wParamdd?

lParamdd?

timedd?

ptdd?

MSGends

; из winuser. h

WM\_CREATEequ1

WM\_ACTIVATEequ6

WM\_SETTEXTequ0Ch

WM\_GETTEXTequ0Dh

WM\_CUTequ300h

WM\_COPYequ301h

WM\_PASTEequ302h

WM\_CLEARequ303h

WM\_UNDOequ304h

WM\_SIZEequ5

WS\_VISIBLEequ10000000h

WS\_CHILDequ40000000h

WS\_EX\_CLIENTEDGE equ200h

ES\_LEFTequ0

ES\_MULTILINEequ4

ES\_AUTOHSCROLLequ80h

ES\_AUTOVSCROLLequ40h

EM\_GETHANDLEequ0BDh

EM\_GETMODIFYequ0B8h

EM\_SETMODIFYequ0B9h

EM\_UNDOequ0C7h

EM\_SETSELequ0B1h

MB\_YESNOequ4

MB\_ICONWARNINGequ30h

IDOKequ1

IDYESequ6

; из commdlg. h

OFN\_PATHMUSTEXISTequ800h

OFN\_FILEMUSTEXISTequ1000h

OFN\_EXPLORERequ80000h

OFN\_OVERWRITEPROMPTequ2

OPENFILENAMEstruc

lStructSizedd?

hwndOwnerdd?

\_hInstancedd?

lpstrFilterdd?

lpstrCustomFilter dd?

nMaxCustFilterdd?

nFilterIndexdd?

lpstrFiledd?

nMaxFiledd?

lpstrFileTitledd?

nMaxFileTitledd?

lpstrInitialDirdd?

lpstrTitledd?

Flagsdd?

nFileOffsetdw?

nFileExtensiondw?

lpstrDefExtdd?

lCustDatadd?

lpfnHookdd?

lpTemplateNamedd?

OPENFILENAMEends

**; user32. inc**

; включаемый файл с определениями функций из user32. dll

;

ifdef \_TASM\_

includelib import32. lib

; имена используемых функций

extrnMessageBoxA: near

extrnDispatchMessageA: near

extrnTranslateMessage: near

extrnGetMessageA: near

extrnLoadIconA: near

extrnUpdateWindow: near

extrnShowWindow: near

extrnCreateWindowExA: near

extrnDefWindowProcA: near

extrnPostQuitMessage: near

extrnRegisterClassExA: near

extrnLoadCursorA: near

extrnLoadMenuA: near

extrnDestroyWindow: near

extrnDialogBoxParamA: near

extrnGetDlgItem: near

extrnSetFocus: near

extrnGetDlgItemTextA: near

extrnSetDlgItemTextA: near

extrnEndDialog: near

extrnLoadAcceleratorsA: near

extrnTranslateAccelerator: near

extrnSendMessageA: near

extrnSetWindowTextA: near

extrnMoveWindow: near

extrnGetClientRect: near

extrnGlobalUnlock: near

; присваивания для облегчения читаемости кода

MessageBoxequMessageBoxA

DispatchMessageequDispatchMessageA

GetMessageequGetMessageA

LoadIconequLoadIconA

CreateWindowExequCreateWindowExA

DefWindowProcequDefWindowProcA

RegisterClassExequRegisterClassExA

LoadCursorequLoadCursorA

LoadMenuequLoadMenuA

DialogBoxParamequDialogBoxParamA

GetDlgItemTextequGetDlgItemTextA

SetDlgItemTextequSetDlgItemTextA

LoadAccelerators equLoadAcceleratorsA

SendMessageequSendMessageA

SetWindowTextequSetWindowTextA

else

includelib user32. lib

; истинные имена используемых функций

extrn\_\_imp\_\_MessageBoxA@16: dword

extrn\_\_imp\_\_DispatchMessageA@4: dword

extrn\_\_imp\_\_TranslateMessage@4: dword

extrn\_\_imp\_\_GetMessageA@16: dword

extrn\_\_imp\_\_LoadIconA@8: dword

extrn\_\_imp\_\_UpdateWindow@4: dword

extrn\_\_imp\_\_ShowWindow@8: dword

extrn\_\_imp\_\_CreateWindowExA@48: dword

extrn\_\_imp\_\_DefWindowProcA@16: dword

extrn\_\_imp\_\_PostQuitMessage@4: dword

extrn\_\_imp\_\_RegisterClassExA@4: dword

extrn\_\_imp\_\_LoadCursorA@8: dword

extrn\_\_imp\_\_LoadMenuA@8: dword

extrn\_\_imp\_\_DestroyWindow@4: dword

extrn\_\_imp\_\_DialogBoxParamA@20: dword

extrn\_\_imp\_\_GetDlgItem@8: dword

extrn\_\_imp\_\_SetFocus@4: dword

extrn\_\_imp\_\_GetDlgItemTextA@16: dword

extrn\_\_imp\_\_SetDlgItemTextA@12: dword

extrn\_\_imp\_\_EndDialog@8: dword

extrn\_\_imp\_\_LoadAcceleratorsA@8: dword

extrn\_\_imp\_\_TranslateAccelerator@12: dword

extrn\_\_imp\_\_SendMessageA@16: dword

extrn\_\_imp\_\_SetWindowTextA@8: dword

extrn\_\_imp\_\_MoveWindow@24: dword

extrn\_\_imp\_\_GetClientRect@8: dword

extrn\_\_imp\_\_GlobalUnlock@4: dword

; присваивания для облегчения читаемости кода

MessageBoxequ\_\_imp\_\_MessageBoxA@16

DispatchMessageequ\_\_imp\_\_DispatchMessageA@4

TranslateMessage equ\_\_imp\_\_TranslateMessage@4

GetMessageequ\_\_imp\_\_GetMessageA@16

LoadIconequ\_\_imp\_\_LoadIconA@8

UpdateWindowequ\_\_imp\_\_UpdateWindow@4

ShowWindowequ\_\_imp\_\_ShowWindow@8

CreateWindowExequ\_\_imp\_\_CreateWindowExA@48

DefWindowProcequ\_\_imp\_\_DefWindowProcA@16

PostQuitMessageequ\_\_imp\_\_PostQuitMessage@4

RegisterClassExequ\_\_imp\_\_RegisterClassExA@4

LoadCursorequ\_\_imp\_\_LoadCursorA@8

LoadMenuequ\_\_imp\_\_LoadMenuA@8

DestroyWindowequ\_\_imp\_\_DestroyWindow@4

DialogBoxParamequ\_\_imp\_\_DialogBoxParamA@20

GetDlgItemequ\_\_imp\_\_GetDlgItem@8

SetFocusequ\_\_imp\_\_SetFocus@4

GetDlgItemTextequ\_\_imp\_\_GetDlgItemTextA@16

SetDlgItemTextequ\_\_imp\_\_SetDlgItemTextA@12

EndDialogequ\_\_imp\_\_EndDialog@8

LoadAccelerators equ\_\_imp\_\_LoadAcceleratorsA@8

TranslateAccelerator equ \_\_imp\_\_TranslateAccelerator@12

SendMessageequ\_\_imp\_\_SendMessageA@16

SetWindowTextequ\_\_imp\_\_SetWindowTextA@8

MoveWindowequ\_\_imp\_\_MoveWindow@24

GetClientRectequ\_\_imp\_\_GetClientRect@8

GlobalUnlockequ\_\_imp\_\_GlobalUnlock@4

Endif

**// winpad95. rc**

// файл ресурсов для программы winpad95. asm

//

// идентификаторы сообщений от пунктов меню

#define IDM\_NEW0x100L

#define IDM\_OPEN0x101L

#define IDM\_SAVE0x102L

#define IDM\_SAVEAS0x103L

#define IDM\_EXIT0x104L

#define IDM\_ABOUT0x105L

#define IDM\_UNDO0x106L

#define IDM\_CUT0x107L

#define IDM\_COPY0x108L

#define IDM\_PASTE0x109L

#define IDM\_CLEAR0x10AL

#define IDM\_SETSEL0x10BL

// идентификаторы основных ресурсов

#define ID\_MENU0x700L

#define ID\_ACCEL 0x701L

#define ID\_ABOUT0x702L

// если есть иконка - можно раскомментировать следующие две строки

// #define ID\_ICON 0x703L

// ID\_ICON ICON "winpad95. ico"

// основное меню

ID\_MENU MENU DISCARDABLE {

POPUP "&File" {

MENUITEM "&New\tCtrl+N", IDM\_NEW

MENUITEM "&Open... \tCtrl+O", IDM\_OPEN

MENUITEM "&Save\tCtrl+S", IDM\_SAVE

MENUITEM "Save &As... \tCtrl+Shift+S", IDM\_SAVEAS

MENUITEM SEPARATOR

MENUITEM "E&xit\tCtrl+Q", IDM\_EXIT

}

POPUP "&Edit" {

MENUITEM "&Undo\tCtrl-Z", IDM\_UNDO

MENUITEM SEPARATOR

MENUITEM "Cu&t\tCtrl-X", IDM\_CUT

MENUITEM "&Copy\tCtrl-C", IDM\_COPY

MENUITEM "&Paste\tCtrl-V", IDM\_PASTE

MENUITEM "&Delete\tDel", IDM\_CLEAR

MENUITEM SEPARATOR

MENUITEM "Select &All\tCtrl-A", IDM\_SETSEL

}

POPUP "&Help" {

MENUITEM "About", IDM\_ABOUT

}

}

// комбинации клавиш

ID\_ACCEL ACCELERATORS DISCARDABLE {

"N", IDM\_NEW, CONTROL, VIRTKEY

"O", IDM\_OPEN, CONTROL, VIRTKEY

"S", IDM\_SAVE, CONTROL, VIRTKEY

"S", IDM\_SAVEAS, CONTROL, SHIFT, VIRTKEY

"Q", IDM\_EXIT, CONTROL, VIRTKEY

"Z", IDM\_UNDO, CONTROL, VIRTKEY

"A", IDM\_SETSEL, CONTROL, VIRTKEY

}

// все эти определения можно заменить на #include <winuser. h>

#define DS\_MODALFRAME0x80L

#define DS\_3DLOOK4

#define WS\_POPUP0x80000000L

#define WS\_CAPTION0xC00000L

#define WS\_SYSMENU0x80000L

#define IDOK1

#define IDC\_STATIC-1

#define IDI\_APPLICATION 32512

#define WS\_BORDER0x800000L

// стандартный диалог "About"

ID\_ABOUT DIALOG DISCARDABLE 0,0,125,75

STYLE DS\_MODALFRAME | DS\_3DLOOK | WS\_POPUP | WS\_CAPTION | WS\_SYSMENU

CAPTION "About Asmpad95"

{

ICONIDI\_APPLICATION, IDC\_STATIC,12,15,21, 20

CTEXT"Asmpad95", IDC\_STATIC,0,30,40,8

CTEXT"Prototype notepad-style editor for Windows 95 written entirely in assembly language",

IDC\_STATIC,45,10,70,45,WS\_BORDER

DEFPUSHBUTTON"OK", IDOK,40,60,40,12

}