МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
імені Семена Кузнеця

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
імені Семена Кузнеця

Лекція № *1*

Змістовний модуль № 1: Системне програмування в Windows

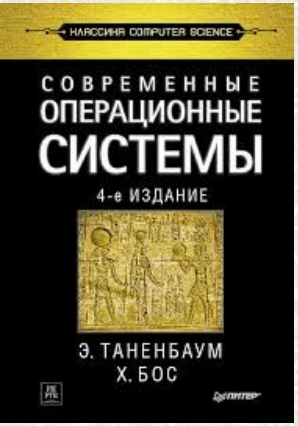
по курсу "**Системне програмування**"

Тема лекції:
Архітектура Windows-додатку

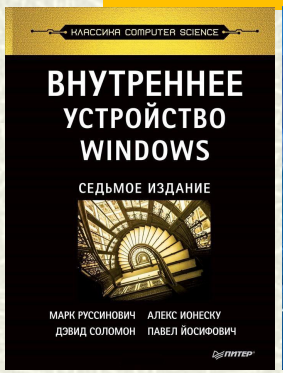
Лектор:
Доцент кафедри Інформаційних систем
кандидат технічних наук, доцент
Голубничий Дмитро Юрійович

04.09.2021

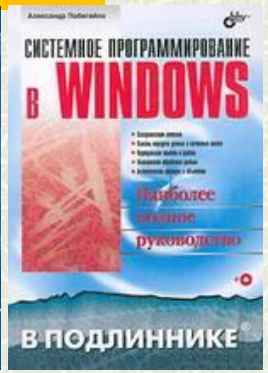
Література


КЛАССИКА COMPUTER SCIENCE
СОВРЕМЕННЫЕ
ОПЕРАЦИОННЫЕ
СИСТЕМЫ
4-е ИЗДАНИЕ
Э. ТАНЕНБАУМ
Х. БОС

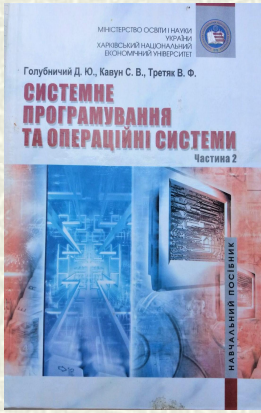
Microsoft
МАСТЕР КЛАСС
ВНУТРЕННЕЕ
УСТРОЙСТВО
Microsoft
Windows


КЛАССИКА COMPUTER SCIENCE
ВНУТРЕННЕЕ
УСТРОЙСТВО
WINDOWS
СЕДЬМОЕ ИЗДАНИЕ
МАРК РУССИНОВИЧ
ДЭВИД СОЛОМОН
АЛЕКС ИОНЕСКУ
ПАВЕЛ ЙОСИФОВИЧ

В. А. Шехонцов
ОПЕРАЦІЙНІ СИСТЕМИ

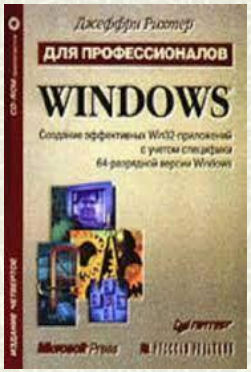

СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ
В WINDOWS
В ПОДЛИННИКЕ

Р.М.Ганеев
Проектирование интерфейса пользователя средствами
Win32 API


Голубничий Д. Ю., Кавун С. В., Третяк В. Ф.
СИСТЕМНЕ ПРОГРАМУВАННЯ
ТА ОПЕРАЦІЙНІ СИСТЕМИ
Частина 2

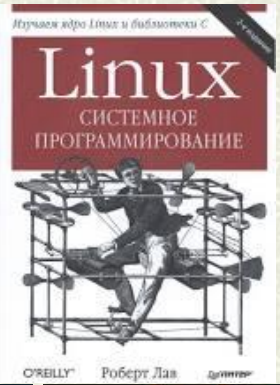
УЧЕБНОЕ ПОСОБИЕ
Проектирование интерфейса пользователя средствами Win32 API
Р. М. Ганеев

БЕСТСЕЛЛЕР НЕ ДЛЯ Ч...
АЗБУКА программирования в Win 32 API
3-е издание
Windows нельзя знать частично...
учебный курс
П.В.РУМЯНЦЕВ

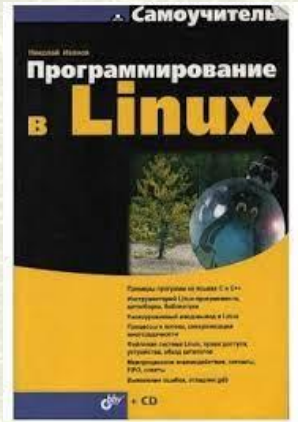

Джеффри Рихтер
ДЛЯ ПРОФЕССИОНАЛОВ
WINDOWS
Создание эффективных Win32-приложений с учетом специфики 64-разрядной версии Windows

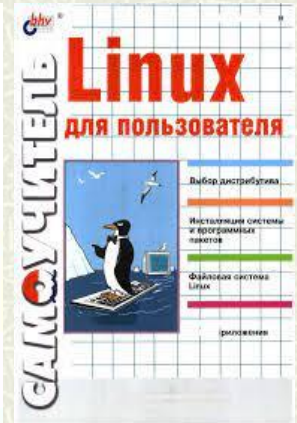
Microsoft
Windows API
справочник системного программиста
DiaSoft
SAMS

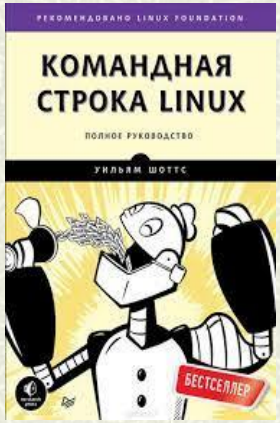
Н. А. Литвиненко
ТЕХНОЛОГИЯ ПРОГРАММИРОВАНИЯ на C++
Win32 API-приложения
УЧЕБНОЕ ПОСОБИЕ

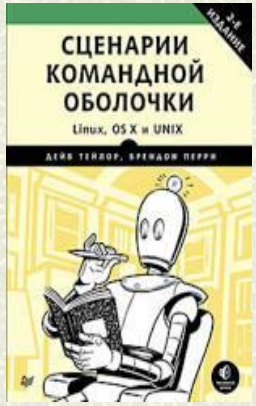

Linux
СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ
O'REILLY
Роберт Лав

Мендель Купер
Искусство программирования на языке сценариев командной оболочки


Самоучитель
Программирование в Linux
+ CD


Linux
для пользователя
САМОУЧИТЕЛЬ


РЕКОМЕНДОВАНО LINUX FOUNDATION
КОМАНДНАЯ СТРОКА LINUX
ПОЛНОЕ РУКОВОДСТВО
УИЛЬЯМ ШОТТС
БЕСТСЕЛЛЕР


СЦЕНАРИИ КОМАНДНОЙ ОБОЛОЧКИ
Linux, OS X и UNIX
ДЕЙВ ТЕЙЛОР, БРЕНДОН ПЕРРИ

Ричард Блум и Кристина Бреснахэн
Командная строка Linux и сценарии оболочки
Библия пользователя
Книга, необходимая для достижения успеха!

Сергей Волков
Ubuntu Linux с нуля

LINUX
на примерах
Практический курс по работе в Linux

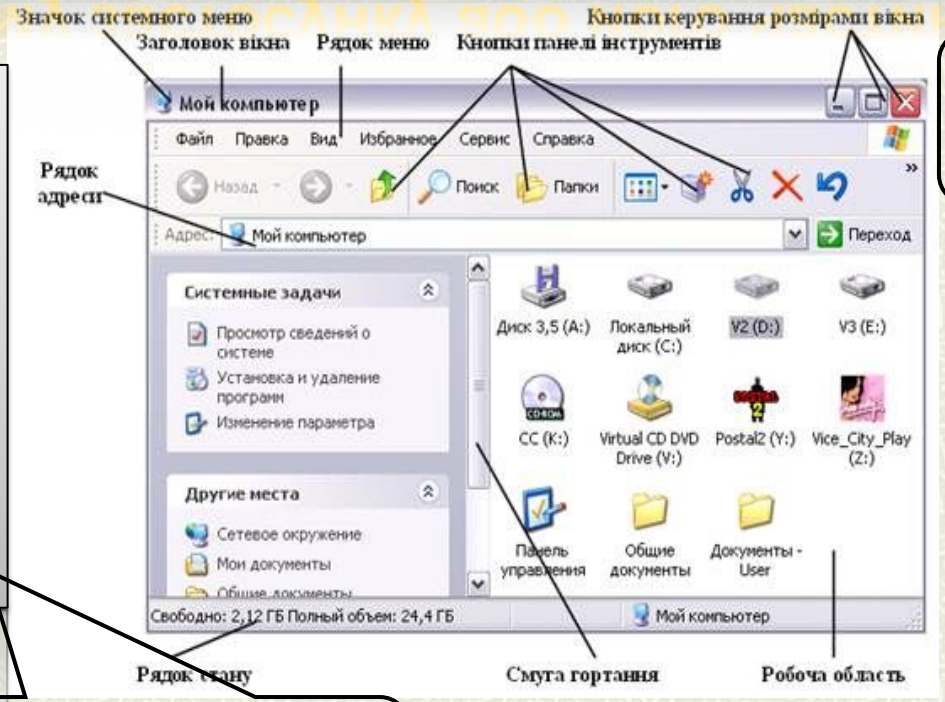
НАВЧАЛЬНІ ПИТАННЯ:

1. Ініціалізація програми.
2. Обробка повідомлень.
3. Файл ресурсів.
4. Меню застосунку.
5. Повідомлення меню.
6. Акселератори.

1. Ініціалізація програми

Вікно - це прямокутна область екрану, яка відповідає будь-якому застосунку або його частини

Вікно - в більшості випадків кінцевий пункт, яким направляються повідомлення



З точки зору користувача

З точки зору операційної системи

Вікно - це об'єкт, атрибути якого (тип, розмір, положення на екрані, вид курсора, меню, значок, заголовок) повинні бути спочатку сформовані, а потім зареєстровані системою

З точки зору програміста

ПРОГРАМУВАННЯ ЗАСТОСУНКУ

Ініціалізація:

Функції та структури:

Етап 1

1. Створення класу вікон із заданими характеристиками
2. Реєстрація віконного класу в Windows
3. Створення головного (батьківського) вікна класу
4. Показ і оновлення вікна на екрані
5. Вхід до циклу повідомлень

WinMain()

WNDCLASS
WNDCLASSEX
RegisterClass()
RegisterClassEx()
CreateWindow()
CreateWindowEx()
ShowWindow()
UpdateWindow()
MSG
GetMessage()
TranslateMessage()
DispatchMessage()

Етап 2

Обробка повідомлень в віконної процедури

WndProc()

DestroyWindow()
DefWindowProc()
PostQuit-Message( )

ФУНКЦІЇ

int APIENTRY WinMain (HINSTANCE hInstance, HINSTANCE hPrevInstance, LPTSTR lpCmdLine, int nCmdShow)

ATOM RegisterClass (CONST WNDCLASS * lpwc)

ATOM RegisterClassEx (CONST WNDCLASSEX * lpwc)

HWND CreateWindow(LPCTSTR lpszAppName, LPCTSTR lpszWindowName, DWORD dwStyle, int x, int y, int nWidth, int nHeight, HWND hwndParent, HMENU hmenu, HINSTANCE hInstance, LPVOID lpvParam);

BOOL ShowWindow (HWND hwnd, int nCmdShow)

BOOL UpdateWindow (HWND *hwnd*)

BOOL GetMessage (LPMSG lpmsg, HWND hWnd, UINT uMsgFilterMin, UINT uMsgFilterMax)

BOOL TranslateMessage (CONST MSG * lpMsg)

LRESULT DispatchMessage (CONST MSG * lpMsg)

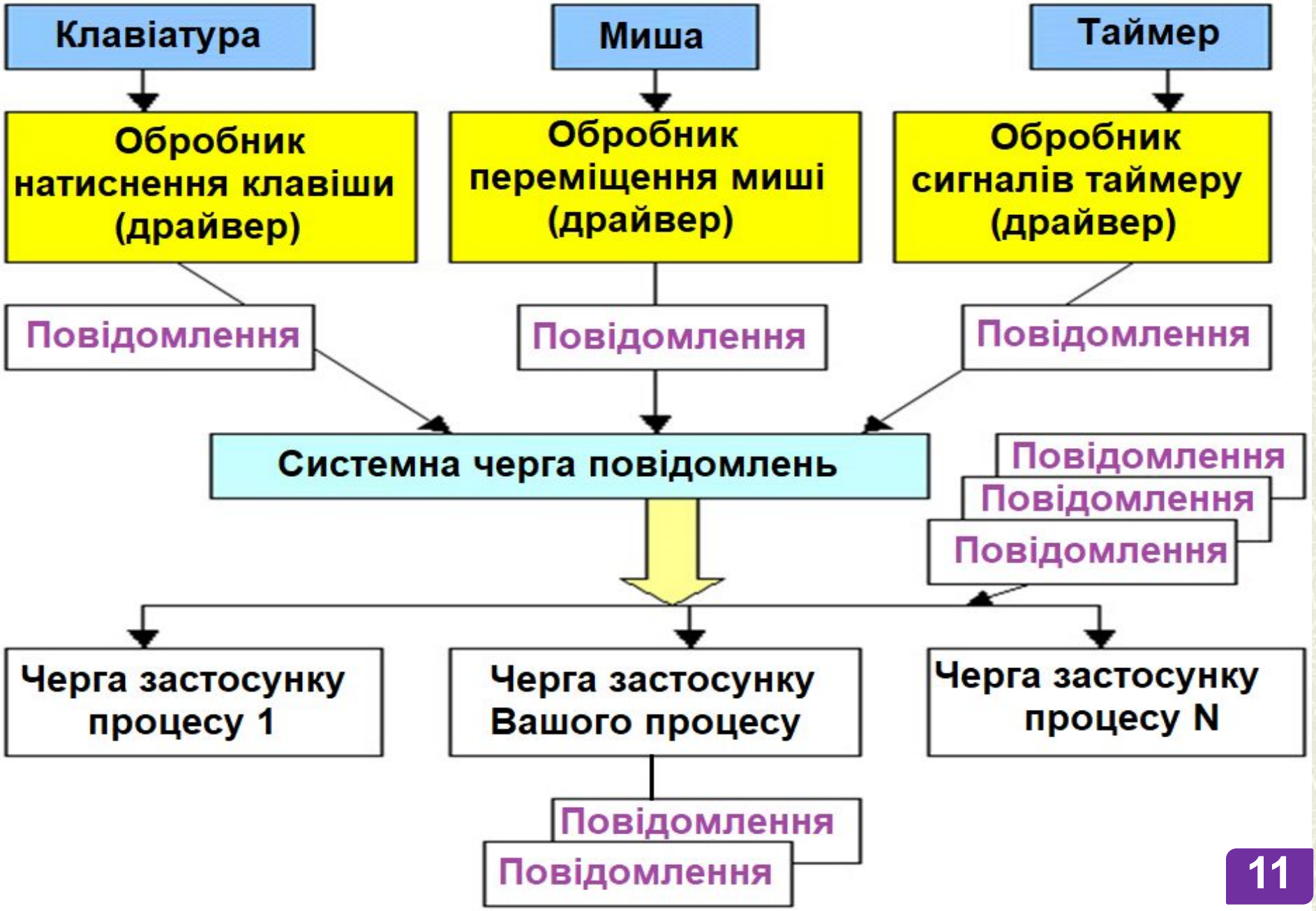
СТРУКТУРИ

```
typedef struct WNDCLASS
{
UINT style; / *стиль вікна
WNDPROC lpfnWndProc; / *покажчик на функцію обробки повідомлень
int cbClsExtra; / *число додаткових байт в кінці структури
int cbWndExtra; / *число додаткових байт за вікном
HINSTANCE hInstance; / *дескриптор додатки
HICON hIcon; / *дескриптор піктограми
HCURSOR hCursor; / *дескриптор курсора
HBRUSH hbrBackground; / *дескриптор кисті
LPCSTR lpszMenuName; / *покажчик на рядок з ім'ям меню
LPCSTR lpszClassName; / * Покажчик на рядок з ім'ям класу вікна
}
```

```
typedef struct MSG
{
HWND hwnd  / * Дескриптор вікна
UINT massge   / * Ідентифікатор повідомлення
WPARAM wParam / * 32-розрядне значення wParam, що передаються функції
WndProc ()
LPARAM lParam / * 32-розрядне значення lParam, що передаються функції
WndProc ()
DWORD time / * Число мілісекунд з моменту запуску
POINT pt   / * Положення курсора миші
}
```

2. Обробка повідомлень

Шлях проходження повідомлень в Windows



Шлях проходження повідомлень в Windows

Microsoft Spy++ - [Messages (All Windows)]

Spy Messages Search View Window Help

```
<00001> 0003026E P WM_PSD_PAGESETUPDLG wPaper:075C wFlags:0 lppsd:00000001
<00002> 0003026E P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00003> 0003026E P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00004> 00010280 P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00005> 0003026E P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00006> 00010280 P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00007> 0003026E P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00008> 0003026E P WM_PSD_PAGESETUPDLG wPaper:075C wFlags:0 lppsd:00000002
<00009> 0003026E P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00010> 0001029A P WM_PSD_PAGESETUPDLG wPaper:0DC0 wFlags:0 lppsd:00000001
<00011> 0001029A P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00012> 0001029A P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00013> 00010280 P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00014> 00010280 P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00015> 0001029A P WM_TIMER wTimerID:8 tmprc:00000000
<00016> 00010280 P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00017> 0001029A P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00018> 0001029A P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00019> 0001029A P WM_PSD_PAGESETUPDLG wPaper:0DC0 wFlags:0 lppsd:00000002
<00020> 0001029A P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00021> 0001029A P WM_PSD_PAGESETUPDLG wPaper:0DC0 wFlags:0 lppsd:00000001
<00022> 0001029A P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00023> 0001029A P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00024> 00010280 P WM_TIMER wTimerID:1 tmprc:00000000
<00025> 00010280 P WM_TIMER wTimerID:25 tmprc:00000000
<00026> 0001029A P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00027> 0001029A P WM_PSD_PAGESETUPDLG wPaper:0DC0 wFlags:0 lppsd:00000001
<00028> 0001029A P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00029> 0001029A P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00030> 00010280 P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00031> 0001029A P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00032> 00010280 P WM_PSD_FULLPAGERECT hDC:00000000 lprcPage:00000000
<00033> 0001029A P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00034> 0001029A P WM_PSD_PAGESETUPDLG wPaper:0DC0 wFlags:0 lppsd:00000002
<00035> 0001029A P WM_PSD_MINMARGINRECT hDC:00000000 lprcMinMargin:00000000
<00036> 000201F0 P message:0x031D [Unknown] wParam:00000003 lParam:00000000
<00037> 0001028A P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000052 lParam:00000045
<00038> 0001013A P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000052 lParam:00000045
<00039> 00010198 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000052 lParam:00000045
<00040> 00010286 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000052 lParam:00000045
<00041> 0001028A P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00042> 0001013A P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00043> 00010196 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000052 lParam:00000045
<00044> 000302BA P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000052 lParam:00000045
<00045> 00010198 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00046> 00010296 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000052 lParam:00000045
<00047> 00010196 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00048> 000202B6 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000052 lParam:00000045
<00049> 000202C0 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000052 lParam:00000045
<00050> 00010286 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00051> 000302BA P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00052> 00010296 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00053> 000202B6 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00054> 000202C0 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00055> 0001028A P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00056> 0001013A P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00057> 00010198 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00058> 00010286 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00059> 00010196 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00060> 000302BA P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
<00061> 00010296 P message:0xC17B [Registered:"WM_MSO_BROADCASTCHANGE"] wParam:00000050 lParam:00000000
```

For Help, press F1

Message Properties

General

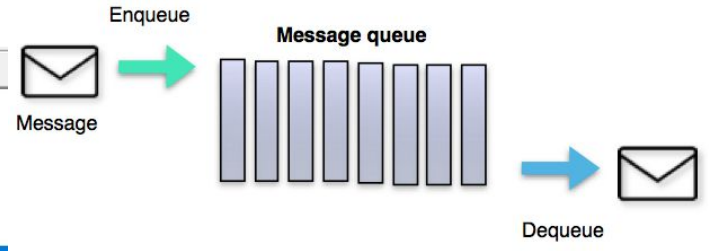
Window Handle: 0003026E

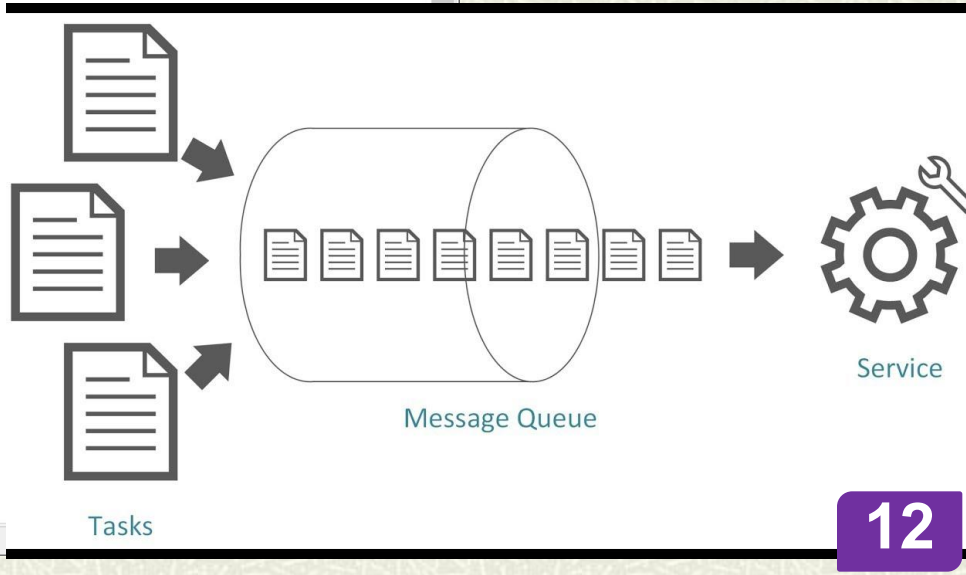
Message: 0402 (Posted)
WM_PSD_MINMARGINRECT

Parameters:

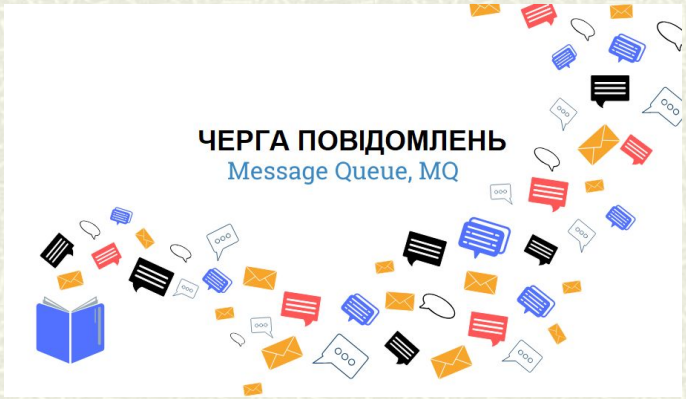
wParam: 00000000
lParam: 00000000 (Pointer to RECT)

Close Справка



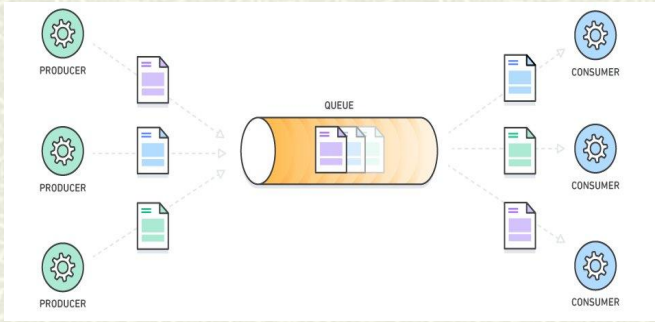


Послідовність обробки повідомлень в Windows



Відбувається апаратна подія

Системна черга повідомлень



Планувальник системи

Черга повідомлень процесу 1

Черга повідомлень процесу 2

Черга повідомлень процесу 3

Функція перехоплення повідомлень процесу 1

Функція перехоплення повідомлень процесу 2

Функція перехоплення повідомлень процесу 3

GetMassage ()
TranslateMessage ()
DispatchMessage ()

GetMassage ()
TranslateMessage ()
DispatchMessage ()

GetMassage ()
TranslateMessage ()
DispatchMessage ()

WndProc ()

WndProc ()

WndProc ()

DefWindowProc ()

DefWindowProc ()

DefWindowProc ()

Віконна процедура (WndProc) - це процедура зворотного виклику (CALLBACK), яка обробляє повідомлення, надіслані їй Windows.

Внутрішня структура віконної процедури:

```
LRESULT CALLBACK WndProc (HWND hWnd, UINT massage, WPARAM wParam,
LPARAM lParam)
{
switch (massage)
   {
      case WM_PAINT:
              // ====== Обробка повідомлення WM_PAINT:
       return 0;
      case WM_DESTROY:
              // ====== Обробка повідомлення WM_DESTROY:
       PostQuitMessage(0) /* функція, яка виставляє повідомлення
                 WM_QUIT, Вказуючи на завершення
виконання процесу */.
       return 0;
      default:
        return (DefWindowProc (hWnd, massage, wParam, lParam));
   }
}
```

Макрос: HANDLE_MSG (HWnd, message, fn);

3.Файл ресурсів

Ресурсами називаються деякі дані, які визначаються ще до початку роботи програми

ОДНОРЯДКОВІ

- BITMAP — растрове зображення
- CURSOR — ресурс курсор
- ICON — ресурс піктограма
- FONT — ресурс шрифт
- MESSAGETABLE — таблиця повідомлень

БАГАТОРЯДКОВІ

- MENU — статичне меню програми
- ACCELERATORS — таблиця гарячих клавіш
- DIALOG — діалогове вікно
- STRINGTABLE — таблиця рядків
- RCDATA — ресурс, визначений користувачем

Однорядкові описувачі ресурсу

BITMAP

файл ресурсу (.rc)*

BITMAPNAME **BITMAP** "*BITMAPFILE.BMP*"

файл програми (. cpp)*

HBITMAP LoadBitmap (HINSTANCE *hInst*, LPCTSTR *lpszBitmap*);

CURSOR

файл ресурсу (.rc)*

CURSORNAME **CURSOR** "*CURSORFILE.CUR*"

файл програми (. cpp)*

HCURSOR LoadCursor (HINSTANCE *hInst*, LPCTSTR *lpszCursor*);

ICON

файл ресурсу (.rc)*

ICONNAME **ICON** "*ICONFILE.ICO*"

файл програми (. cpp)*

HICON LoadIcon (HINSTANCE *hInst*, LPCTSTR *lpszIcon*);

Видалення об'єкта

BOOL DeleteObject (HGDIOBJ *hGdiObject*);

Багаторядкові описувачі ресурсу

STRINGTABLE

файл ресурсу (.rc)*

```
STRINGTABLE
{
IDS_STRING1, "рядок 1 "
IDS_STRING2, "рядок 2 "
IDS_STRING3, "рядок 3 "
}
```

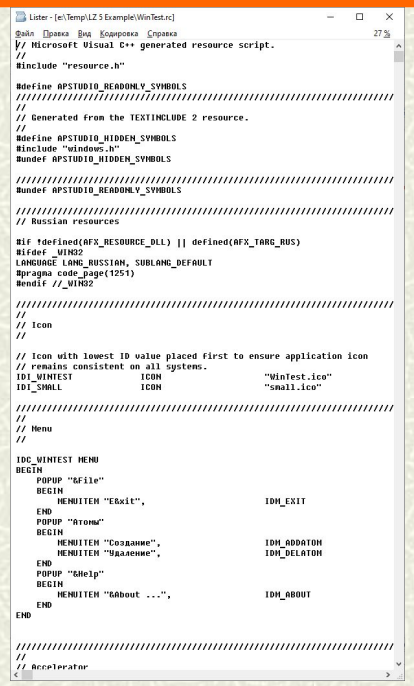
файл програми (. cpp)*

int **LoadString** (HINSTANCE *hInst*, UINT *uID*, LPTSTR *lpBuffer, int nBuffer);

```
//////////////////////////////////////////////////////////////////////////////
//
// String Table
//

STRINGTABLE
BEGIN
    IDS_APP_TITLE           "WinTest"
    IDC_WINTEST             "WINTEST"
END

#endif    // Russian resources
//////////////////////////////////////////////////////////////////////////////
```



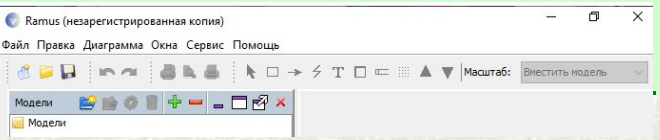
4.Меню застосунку

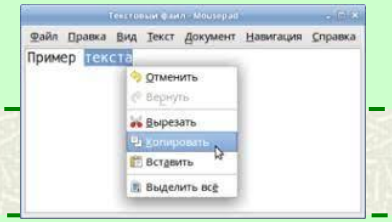
Багаторядкові описувачі ресурсу

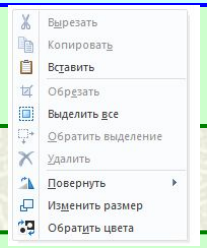
MENU

Види меню

- Головне меню вікна
- Спливаюче (popup) меню
- Контекстне меню







За способом створення

- **Статичне меню** *(з використанням файлу ресурсів)*
- **Динамічне меню** *(без використання файлу ресурсів)*

Багаторядкові описувачі ресурсу

Статичне меню *(з використанням файлу ресурсів)*

файл ресурсу (.rc)*

1. Багаторівнева деревоподібна структура меню

MenuName ***MENU*** [Параметри] // це - головне меню
{
// Опис усіх popup-меню і елементів меню другого рівня
}

2. Спливаючі меню

POPUP "Ім'я" [, Параметри] // - опис popup-меню
{
// Опис усіх popup-меню і елементів чергового рівня
}

```
//////////////////////////////////////////////////////////////////////////////
//
// Menu
//

IDC_WINTEST MENU
BEGIN
    POPUP "&File"
    BEGIN
        MENUITEM "E&xit",                       IDM_EXIT
    END
    POPUP "Атомы"
    BEGIN
        MENUITEM "Создание",                    IDM_ADDATOM
        MENUITEM "Удаление",                    IDM_DELATOM
    END
    POPUP "&Help"
    BEGIN
        MENUITEM "&About ...",                  IDM_ABOUT
    END
END
```

3. Кінцевий елемент меню

MENUITEM "Ім'я", MenuID [, Параметри]

Примітка. Якщо замість імені меню вікна записано слово SEPARATOR (без лапок), на місці елемента меню з'являється горизонтальна лінія

4. Додавання посилання на ім'я меню в структурі WNDCLASS

WndClass.lpszMenuName = " MenuName "

Багаторядкові описувачі ресурсу

Статичне меню *(з використанням файлу ресурсів)*

Параметри, що описують елемент меню в файлі ресурсів

Прапор	Значення
CHECKED	Поруч з ім'ям елемента може відображатися невеликий значок, який свідчить про те, що відповідний прапор встановлений
ENABLED	Елемент меню доступний
DISABLED	Елемент меню недоступний, але відображається як звичайний
GRAYED	Елемент меню недоступний і відображається сірим кольором
MENUBREAK	Горизонтальні меню розміщують такі елементи в новому рядку, а вертикальні - в новому стовпці
MENUBARBREAK	Те ж, що і попереднє, але в разі вертикального меню стовпці поділяються вертикальною лінією

Багаторядкові описувачі ресурсу

Статичне меню *(з використанням файлу ресурсів)*

Наприклад. Спробуємо створити опис невеликого меню.

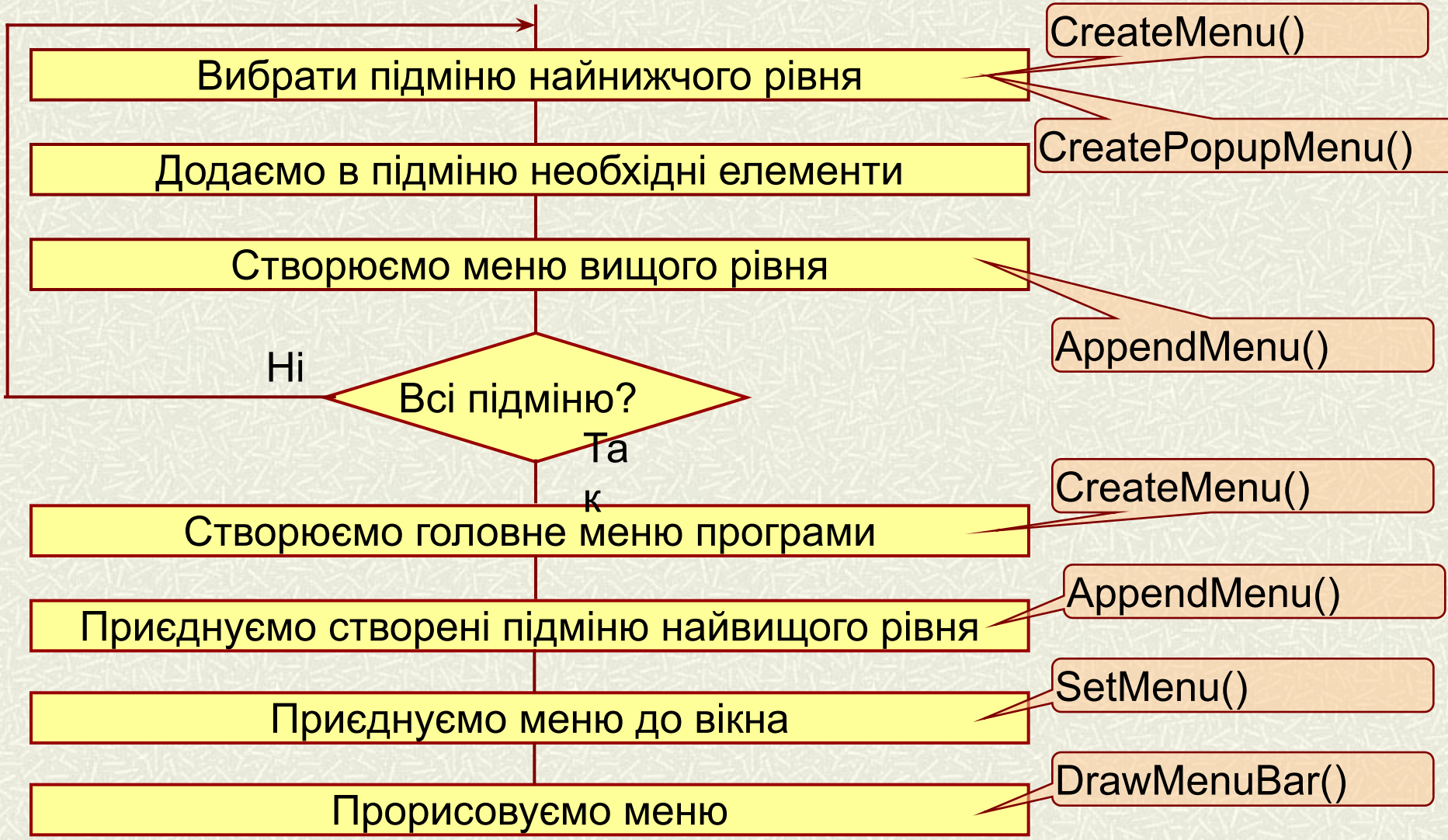
Горизонтальне меню дозволить вибирати підміню "File", "Examples" І кінцевий елемент "Help".

Підменю"File" буде містити елементи "Open" І "Exit", що розділені горизонтальною лінією, а підміню"Examples" - кілька кінцевих елементів.

```
MyMenu MENU
{
POPUP "& File"
{
MENUITEM "& Open", 101
MENUITEM SEPARATOR
MENUITEM "E & xit", 102
}
POPUP "& Examle"
{
POPUP "Example1"
{
      MENUITEM "1 & 1", 103
      MENUITEM "1 & 2", 104
}
POPUP "Example2"
{
      MENUITEM "2 & 1", 105
      MENUITEM "2 & 2", 106
}
}
MENUITEM "& Help", 111
}
```

Багаторядкові описувачі ресурсу

Динамічне меню *(без використання файлу ресурсів)*



Алгоритм створення динамічного меню

Багаторядкові описувачі ресурсу

Контекстне меню

Для відображення і вибору рядків контекстного меню

BOOL ***TrackPopupMenu*** (HMENU *hMenu*, UINT *uFlags*, int *x,* int *y,* int *nReserved*, HWND *hwnd*, CONST RECT **prcRect*);

Константа	*пояснення*
TPM_CENTERALIGN	Центр меню по горизонталі збігається з x
TPM_LEFTALIGN	Лівий край меню збігається з x
TPM_RIGHTALIGN	Правий край меню збігається з x
TPM_BOTTOMALIGN	Нижній край меню збігається з y
TPM_TOPALIGN	Верхній край меню збігається з y
TPM_VCENTERALIGN	Центр меню по вертикалі збігається з y
TPM_NONOTIFY	Не посилати повідомлення про вибір рядка
TPM_RETURNCMD	Повертати ідентифікатор обраної команди
TPM_LEFTBUTTON	Простежує ліву кнопку миші
TPM_RIGHTBUTTON	Простежує праву кнопку миші

5.Повідомлення меню

ПОВІДОМЛЕННЯ МЕНЮ

WM_MENUSELECT

Формується при зміні підсвіченого елемента меню покажчиком миші

wParam

LOWORD (wParam)	HIWORD(wParam)
0 … 15	16 … 31

якщо підсвічений елемент є кінцевим і не тягне за собою виклик popup-меню, то містить ідентифікатор елемента меню

якщо підсвічений елемент при виборі тягне за собою виклик popup-меню, то містить номер (індекс) цього елемента в тому меню, в якому воно знаходиться

Характеристики підсвіченого елемента меню (Код нотифікації)

прапор	опис
MF_BITMAP	Замість рядка в якості елемента меню застосовується bitmap
MF_CHECKED	Елемент відзначається (зі «значком»)
MF_DISABLED	елемент заборонений
MF_GRAYED	Елемент заборонений і відображається сірим кольором
MF_HILITE	елемент підсвічений
MF_MOUSE-SELECT	Елемент обраний мишею
MF_OWNER-DRAW	За промальовування елемента відповідає не система, а програма
MF_POPUP	Елемент викликає поява popip-меню нижчого рівня
MF_SYSMENU	Елемент з системного меню

lParam

Хендл того меню, якому належить підсвічений елемент

ПОВІДОМЛЕННЯ МЕНЮ

WM_COMMAND

Формується при виборі (натисканні клавіші "Enter") звичайного пункту меню

wParam

0 LOWORD (wParam) 15	16 HIWORD(wParam) 31
ідентифікатор обраного елемента меню	**дорівнює нулю (що означає від меню)**

lParam

завжди NULL !!!

WM_SYSCOMMAND

Формується при виборі (натисканні клавіші "Enter") пункту системного меню

wParam

Запрос на системну команду. Починається з префіксаSC_

lParam

0 LOWORD (lParam) 15	16 HIWORD(lParam) 31
x - координата курсора миші; 0 - якщо миша не використовується	y - координата курсора миші; 0 - якщо миша не використовується

ПОВІДОМЛЕННЯ МЕНЮ

WM_INITMENU

Формується перед відображенням головного вікна

wParam

хендл меню

lParam

не використовується

Примітка. *Це повідомлення надсилається при натисканні кнопки миші в рядку меню або при натисканні клавіші меню*

WM_INITMENUPOPUP

Формується перед відображенням спливаючого меню

wParam

хендл підміню

lParam

0 LOWORD (lParam) 15	16 HIWORD(lParam) 31
індекс спливаючого меню в меню верхнього рівня	**1 - системне меню;** **0 - звичайне меню.**

Примітка. Це повідомлення дає прикладної задачі можливість змінювати стан елементів меню перед відображенням спливаючого меню

ФУНКЦІЇ ДЛЯ УПРАВЛІННЯ МЕНЮ

Зміна стану елемента пункту меню

BOOL ***EnableMenuItem*** (HMENU *hMenu*, UINT *uItem*, UINT *uFlags*);

Зміна напису елемента пункту меню

BOOL ***ModifyMenu*** (HMENU *hMenu*, UINT *uItem*, UINT *uFlags*, UINT *uNewItem*, LPCTSTR *lpszNewItem*);

Видалення елемента з меню і звільнення всіх пов'язаних з ним ресурсів

BOOL ***DeleteMenu*** (HMENU *hMenu*, UINT *uPosition*, UINT *uFlags*);

Видалення елемента з меню без звільнення всіх пов'язаних з ним ресурсів

BOOL ***RemoveMenu*** (HMENU *hMenu*, UINT *uPosition*, UINT *uFlags*) ;

Повне видалення (руйнування) меню

BOOL ***DestroyMenu*** (HMENU *hMenu*) ;

Повертає хендл головного меню вікна

HMENU ***GetMenu*** (HWND *hWnd*) ;

Повертає хендл спливаючого меню

HMENU ***GetSubMenu*** (HMENU *hMenu,* int *nPos*) ;

РЯДОК СТАНУ

Функції для роботи з рядком стану оголошені у файлі ***commctrl.h***

```
#pragma comment (lib, "comctl32.lib")
```

Ініціалізація бібліотеки загальних елементів управління

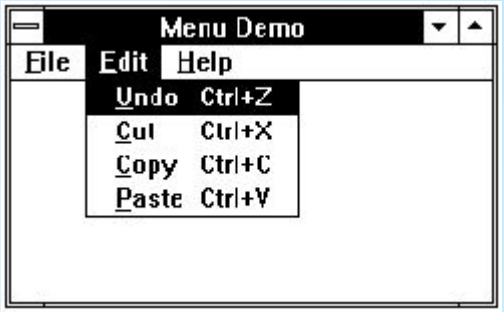
void ***InitCommonControls*** (VOID);

Створення вікна рядка стану

HWND ***CreateStatusWindow***(LONG *style*, LPCTSTR *lpszText*, HWND *hwndParent*, UINT *wID*);

Повідомлення	Опис
WM_LBUTTONDBLCLK	Ліва кнопка миші натиснута вдруге протягом часу реєстрації подвійного клацання
WM_LBUTTONDOWN	Натиснута ліва кнопка миші
WM_LBUTTONUP	Відпущена ліва кнопка миші
WM_MOUSEMOVE	Перенесуть курсор миші
WM_NCMOUSEMOVE	Курсор миші пересунувся всередині некліентской області вікна
WM_RBUTTONDBLCLK	Права кнопка миші натиснута вдруге протягом часу реєстрації подвійного клацання
WM_RBUTTONDOWN	Натиснута права кнопка миші
WM_RBUTTONUP	Відпущена права кнопка миші
WM_SETCURSOR	Повинна бути виконана зміна курсору миші

6. Акселератори



Комбінації клавіш, які при натисканні автоматично вибирають відповідний їм елемент меню (навіть в тих випадках, коли воно не активно і не відображається), називаються ***акселераторами****.*

Віртуальна клавіша *- це системно-незалежний код, визначений для основного набору службових клавіш.*

Несистемною клавішею називається будь-яка клавіша, що натискається без одночасно натискання клавіші Alt.

файл ресурсу (.rc)*

```
TableName ACCELERATORS
{Key1, MenuID1 [,тип] [,параметр]
. . . . . .
Keyn, MenuIDn [,тип] [, параметр]
}
```

```
//////////////////////////////////
//
// Accelerator
//

IDC_WINTEST ACCELERATORS
BEGIN
    "?", IDM_ABOUT, ASCII,  ALT
    "/", IDM_ABOUT, ASCII,  ALT
END
```

Повідомлення	Опис
WM_KEYUP	Повідомляє вікно з фокусом, що несистемна клавіша відпущена.
WM_KEYDOWN	Повідомляє вікно з фокусом, що була натиснута несистемна клавіша.
WM_CHAR	Повідомляє вікно про код символу натиснутоюклавіші коли викликається функція TranslateMessage ()

Завантаження таблиці акселераторів з файлу ресурсів

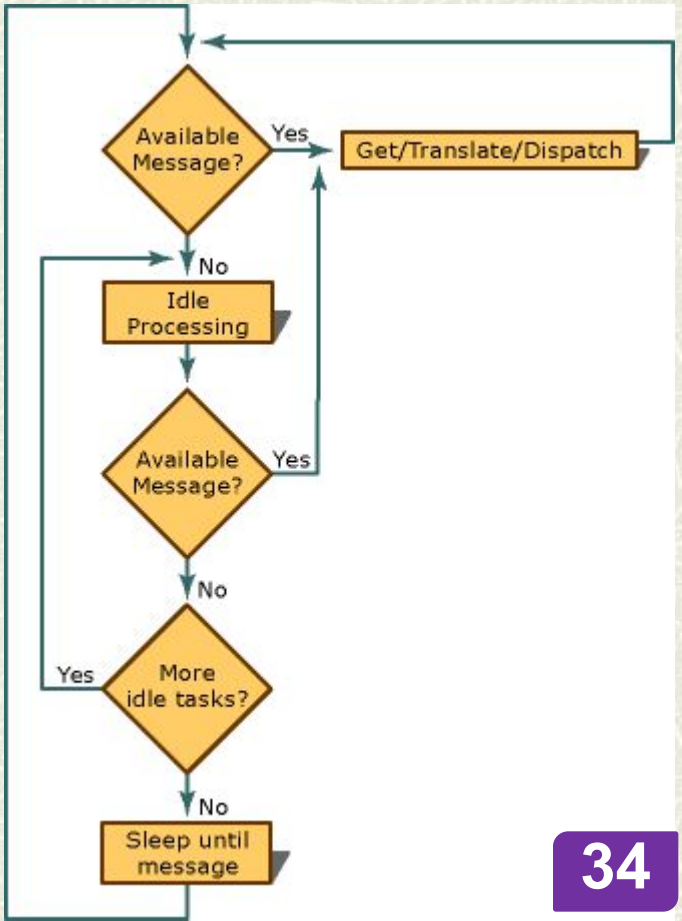
HACCEL ***LoadAccelerators*** (HINSTANCE *hInst*, LPCTSTR *lpszTableName*);

Переклад акселератор клавіатура (wm_KeyUp, wm_KeyDown) в повідомлення команд меню

int ***TranslateAccelerator*** (HWND *hWnd*, HACCEL *hAccTable*, LPMSG *lpMsg*);

Цикл обробки повідомлень

```
while (GetMessage (& msg, NULL, 0, 0))
    {
    if          (!          Accel          ||!
TranslateAccelerator   (hWnd,   hAcc,   &
msg);
        {
        TranslateMessage (& msg);
           DispatchMessage (& msg);
        }
    }
```



Дякую за увагу!