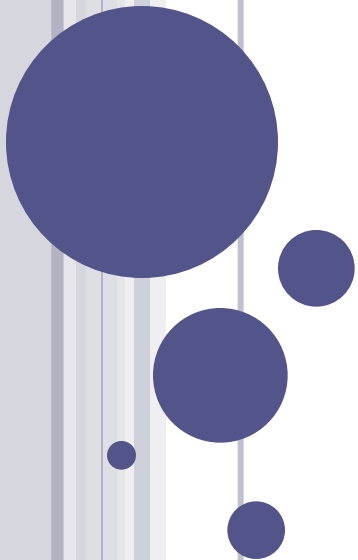


ЛЕКЦІЯ 6

Система повідомлень ОС Windows

Операційні системи
доц. Сінельнікова Т.Ф.



СИСТЕМА ПОВІДОМЛЕНЬ ОС WINDOWS

Обробка повідомлень лежить в основі роботи ОС сімейства Windows, а також додатків, що працюють під їх управлінням. Система і додатки виробляють повідомлення у відповідь на кожну подію, що відбувається в Windows. Повідомлення дозволяють Windows виконувати кілька додатків одночасно. У системах Windows 95 і в ОС сімейства Windows NT кожен потік або процес отримує свою власну чергу повідомлень, що дозволяє їм працювати незалежно. У більш ранніх версіях Windows всі програми поділяли одну чергу повідомлень. У цих версіях Windows-додаток повинен був повертати управління Windows якомога швидше, щоб дати можливість іншим програмам обробляти повідомлення.

СИСТЕМА ПОВІДОМЛЕНЬ ОС WINDOWS

Основою механізму повідомлень ОС Windows є **потік повідомлень**. Windows виробляє повідомлення у відповідь на кожну апаратну подію, таке як натискання користувачем клавіші на клавіатурі або переміщення миші. Система передає ці повідомлення у відповідну чергу повідомлень потоку. Кожен потік в системі обробляє повідомлення у власній черзі повідомлень. Якщо повідомлення призначене для конкретного потоку, воно поміщається в чергу повідомлень цього потоку. Деякі повідомлення відносяться до всієї системи або призначені для декількох потоків. Ці повідомлення поміщаються в чергу відповідних потоків. **Повідомлення** це структура даних, яка визначена наступним чином:

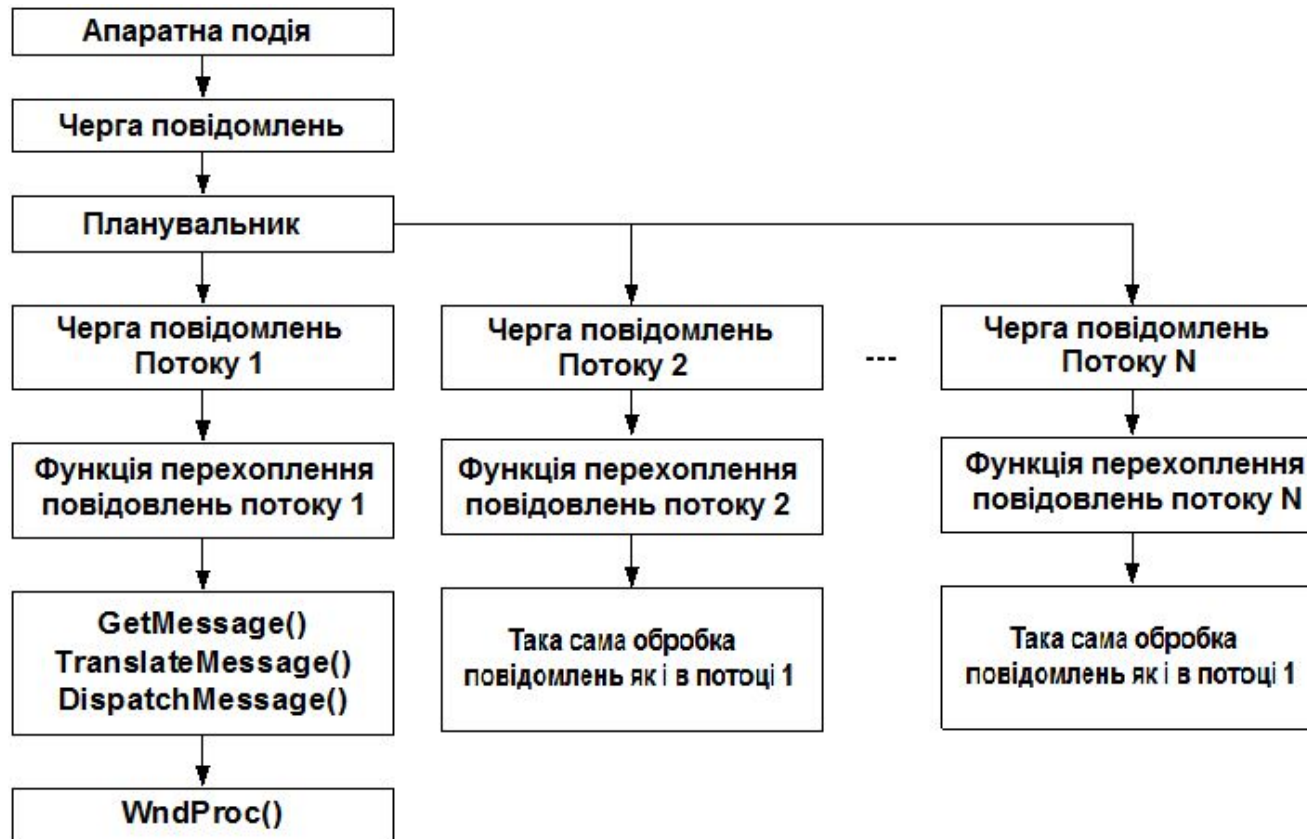
СИСТЕМА ПОВІДОМЛЕНЬ ОС WINDOWS

```
typedef struct tagMSG {
    HWND    hwnd;      /* дескриптор окна */
    UINT    message;  /* идентификатор сообщения */
    WPARAM wParam;    /* значение wParam */
    LPARAM lParam;    /* значение lParam */
    DWORD   time;     /* число миллисекунд с момента запуска */
    POINT   pt;       /* положение курсора мыши в экранных координатах
*/
} MSG;
```

Дані повідомлення містять *дескриптор вікна (hwnd)*, *закодований тип повідомлення (message)*, *дані wParam* і *lParam*, які будуть передані функції головного вікна програми, *позначку часу відправлення повідомлення time* (в мілісекундах після запуску Windows) і *структуру POINT (pt)*, що містить координати x і y курсора миші під час відправки повідомлення.

СИСТЕМА ПОВІДОМЛЕНЬ ОС WINDOWS

На малюнку показана спрощена схема для одного потоку повідомлень і для одного повідомлення, що проходить через процедуру обробки повідомлення даного потоку. Для інших потоків цей процес відбувається аналогічно.



СИСТЕМА ПОВІДОМЛЕНЬ ОС WINDOWS

Повідомлення з'являються не тільки при виникненні апаратних подій (натискання на клавіатурі клавіши або рух миші). Часто повідомлення використовуються для управління програмою або відправлення будь-якої інформації іншим програмам. Потoki можуть обмінюватися даними, відправляючи одне одному повідомлення. Повідомлення можуть поміщатися в чергу з використанням функції **PostMessage** (це називається *виставленням повідомлення*) або за допомогою функції **SendMessage** відправлятися прямо в чергу повідомлення для негайної обробки.

СИСТЕМА ПОВІДОМЛЕНЬ ОС WINDOWS

Також можна здійснювати відправлення повідомлень між програмами. В ОС Windows передбачена функція **RegisterWindowMessage**, що дозволяє динамічно виробляти унікальні ідентифікатори повідомлень. Ці ідентифікатори отримують такі послідовні номери, щоб вони не збігалися з номерами повідомлень, що застосовані в програмах з іншого набору програм. Стандартний спосіб обміну даними між програмами полягає у використанні динамічного обміну даними (**DDE - dynamic data exchange**). **DDE** - це ряд повідомлень, які визначені для використання в якості стандарту зв'язку.

ДЕЯКІ ФУНКЦІЇ ДЛЯ РОБОТИ З ПОВІДОМЛЕННЯМИ

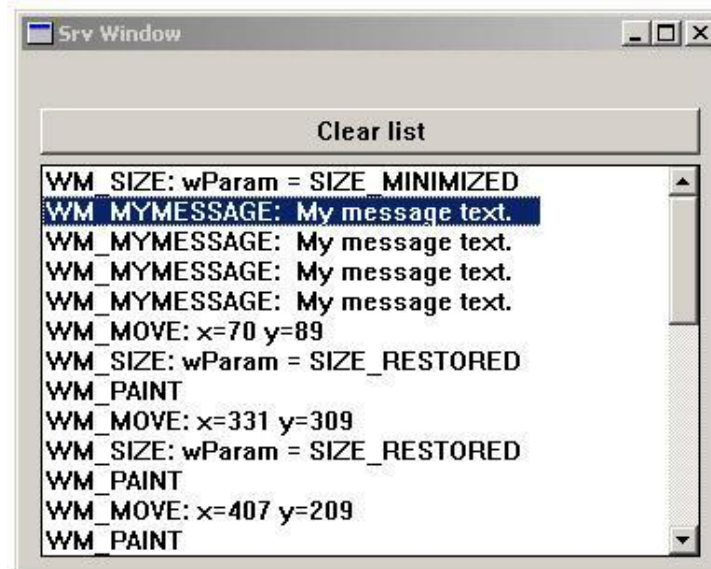
Функція	Назначение
BroadcastSystemMessage	Відправляє повідомлення списку одержувачів, таких як додатки, драйвери і т. д.
CallMsgFilter	Активізує функцію фільтрації повідомлень (функцію перехоплення повідомлень)
CallNextHookEx	Передає інформацію наступній процедурі перехоплення повідомлень у поточному ланцюжку функцій перехоплення повідомлень
CallWindowProc	Передає параметри повідомлення функції обробки повідомлень
DefWindowProc	Передбачена за замовчуванням функція обробки повідомлень
DispatchMessage	Відправляє повідомлення Windows функції WndProc
ExitWindows	Припиняє сеанс роботи поточного користувача
ExitWindowsEx	Припиняє сеанс роботи поточного користувача, зупиняє комп'ютер або перезапускає комп'ютер
GetMessage	Вибирає повідомлення з черги повідомлень додатку
GetMessageExtraInfo	Вибирає додаткову інформацію про останнє обране повідомлення
GetMessagePos	Вибирає позицію курсора на момент відправки повідомлення
GetMessageTime	Вибирає час відправки повідомлення
InSendMessage	Визначає, чи було повідомлення надіслано з використанням функції SendMessage
InSendMessageEx	Визначає, чи обробляє поточна віконна процедура повідомлення, яке було надіслано з іншого потоку

ДЕЯКІ ФУНКЦІЇ ДЛЯ РОБОТИ З ПОВІДОМЛЕННЯМИ

Функція	Назначення
PeekMessage	Перевіряє чергу повідомлень на наявність повідомлення
PostMessage	Поміщає повідомлення в чергу повідомлень і повертається без очікування
PostQuitMessage	Зупиняє додаток
PostThreadMessage	Поміщає повідомлення в чергу повідомлень потоку і повертається без очікування
RegisterWindowMessage	Створює новий, унікальний номер повідомлення Windows
ReplyMessage	Відповідає на повідомлення, яке було надіслано з допомогою функції SendMessage
SendMessage	Відправляє повідомлення безпосередньо функції обробки повідомлень вікна і чекає відповіді
SendMessageCallback	Відправляє повідомлення і негайно виконує повернення, дозволяючи функції зворотного виклику обробити відповідь
SendMessageTimeout	Відправляє повідомлення і чекає відповіді до закінчення тайм-ауту
SendNotifyMessage	Відправляє повідомлення і негайно виконує повернення
SetMessageExtraInfo	Встановлює додаткову інформацію повідомлення для поточного потоку
SetWindowsHookEx	Встановлює функцію фільтрації повідомлень Windows
TranslateMessage	Генерує повідомлення WM_CHAR , WM_SYSCHAR , WM_DEADCHAR и WM_SYSDEADCHAR при отриманні кода віртуальної клавіші
UnhookWindowsHookEx	Видаляє функцію перехоплення повідомлення з системи WaitMessage та передає управління іншим потокам на час очікування повідомлення

ПРИКЛАД ПРОГРАМИ

Приклад, який демонструє застосування функцій **SendMessage**, **GetMessage**, **TranslateMessage**, **DispatchMessage**, розглянемо програму *MsgDemo*, яка здійснює відправку повідомлень з головного вікна програми іншого вікна, і ці повідомлення управляють другим вікном. Друге вікно отримує ці повідомлення, обробляє їх і виконує дії, які повинні виконуватися при отриманні відповідних повідомлень.



ПРИКЛАД ПРОГРАМИ

На головному вікні програми розташовані кнопки *Close*, *Maximize*, *Minimize*, *Send Custom Message*. Натискання на кнопки *Close*, *Maximize*, *Minimize* ініціює відправку повідомлення в інше вікно, яке їх обробляє і здійснює відповідну дію - закривається, максимізується, мінімізується. Кнопка *Send Custom Message* дозволяє відправити довільне повідомлення, яке набирається в рядку під кнопками замість напису "My message text". Також текст цього повідомлення відображається і нижче для показу, що повідомлення надіслано. Друге вікно при отриманні такого повідомлення відображає його текст разом з іншими повідомленнями, які йому надійшли.

Вихідний код програми приведений в навчальному посібнику.