

III. Lucrul cu resurse

Resurse

- date în format binar, incluse în fișierul executabil
- nu sunt definite în codul sursă
 - dar există funcții pentru manipularea lor
 - sunt gestionate direct de sistemul Windows
- tipuri
 - standard (predefinite)
 - definite de utilizator

Resurse standard

- imagini
 - icon-uri, bitmap-uri, cursoare pentru mouse
- meniuri
- acceleratori
 - combinații de taste pentru declanșarea unor acțiuni
- dialoguri
- șiruri de caractere etc.

Specificarea resurselor

- cod sursă - fișier cu extensia .rc
 - resursele de tip imagine au asociate fișiere în format BMP sau ICO
- după compilare - fișier cu extensia .res
- fișierul sursă poate fi modificat cu un editor de text
- Visual Studio - editorul de resurse
 - definire vizuală a resurselor

Utilizarea resurselor

- în program, resursele trebuie încărcate cu ajutorul unor funcții specifice
- parametri uzuali
 - *handle* către instanța curentă a aplicației
 - primit ca parametru de funcția WinMain
 - numele resursei
 - obținut din identificatorul său din fișierul de resurse
 - se utilizează macroul MAKEINTRESOURCE

Imagini

- funcții pentru încărcare

```
HICON LoadIcon(HINSTANCE hInst,  
    LPCTSTR lpIconName);
```

```
HBITMAP LoadBitmap(HINSTANCE  
    hInst, LPCTSTR lpBitmapName);
```

```
HCURSOR LoadCursor(HINSTANCE  
    hInst, LPCTSTR lpCursorName);
```

Exemple de definire

IDI_ICON1	ICON	"i.ico"
IDI_BITMAP1	BITMAP	"b.bmp"
IDI_CURSOR1	CURSOR	"c.cur"

Șiruri de caractere

- încărcare

```
int LoadString(HINSTANCE hInst,  
    UINT uID, LPTSTR lpBuffer,  
    int nBufferMax);
```

- preia resursa și o depune în șirul lpBuffer
- cel mult nBufferMax caractere
- pot apărea probleme la utilizarea UNICODE

Exemplu de definire

```
STRINGTABLE
```

```
BEGIN
```

```
    IDS_STRING1    "First string"
```

```
    IDS_STRING2    "Second string"
```

```
END
```

Dialoguri

- ferestre speciale, destinate preluării de date de la utilizator
- un dialog este creat când este necesar și distrus când nu mai este nevoie de el
- pentru comunicare, în dialog sunt incluse controale

Tipuri de dialoguri

- modale
 - fereastra părinte nu este accesibilă decât după distrugerea dialogului
 - cele mai utilizate
- nemodale (*modeless*)
 - fereastra părinte poate fi accesată și în timp ce dialogul este activ

Crearea dialogurilor modale

```
INT_PTR DialogBox(HINSTANCE  
    hInst, LPCTSTR lpTemplate,  
    HWND hWndParent, DLGPROC  
    lpDialogFunc);
```

- parametri specifici
 - hWndParent - fereastra părinte a dialogului
 - lpDialogFunc - funcția de tratare a mesajelor

Crearea dialogurilor nemodale

```
HWND CreateDialog(HINSTANCE  
    hInst, LPCTSTR lpTemplate,  
    HWND hWndParent, DLGPROC  
    lpDialogFunc);
```

- parametrii au aceeași semnificație ca în cazul funcției `DialogBox`
- tipul returnat este diferit

Terminarea unui dialog

- apelul unei funcții specializate
 - atunci când dialogul și-a îndeplinit sarcina
- dialoguri modale

```
BOOL EndDialog (HWND hDlg,  
    INT_PTR nResult) ;
```

- dialoguri nemodale

```
BOOL DestroyWindow (HWND hWnd) ;
```

Exemplu de definire

```
IDD_DIALOG1 DIALOGEX 0, 0, 336, 133
STYLE DS_MODALFRAME | WS_SYSMENU
CAPTION "Dialog"
FONT 8, "MS Shell Dlg", 400, 0, 0x1
BEGIN
    DEFPUSHBUTTON "OK",IDOK,179,7,50,14
    PUSHBUTTON "Cancel",IDCANCEL,179,24,50,14
    EDITTEXT IDC_EDIT1,65,44,79,14
END
```

Tratarea mesajelor

- funcția de tratare a mesajelor în cadrul dialogului
 - similară celei a ferestrei principale
 - nu mai este necesară tratarea implicită a mesajelor (prin *DefWindowProc*)
- tipuri de mesaje
 - mesaje generale Windows
 - mesaje specifice controalelor

Mesaje generale Windows

```
switch (message) {  
    case WM_INITDIALOG:  
        // creare dialog  
    case WM_PAINT:  
        // afisare dialog  
    // alte mesaje  
}
```

Mesaje specifice controalelor

```
case WM_COMMAND:
    switch (LOWORD(wParam)) {
        case IDOK:
            // apasare buton OK
        case IDCANCEL:
            // apasare buton Cancel
        // alte mesaje
    }
```

De unde vin informațiile?

- WM_COMMAND - mesajul general prin care se preiau acțiunile specifice
- wParam - parametru primit de funcția de tratare a mesajelor
- LOWORD - preia cei 2 octeți mai puțin semnificativi ai argumentului
 - similar - HIWORD
- uneori trebuie preluat HIWORD(wParam)

Evenimente generate de controale

Exemple

- edit box
 - EN_CHANGE - schimbarea textului
- combo box
 - CBN_SELCHANGE - selectarea unui element
- button
 - BN_CLICKED - apăsarea butonului

Transferul datelor (1)

- dialogul trebuie să transfere date spre/dinspre controalele incluse
- se realizează prin transmiterea de mesaje către controlul dorit

```
LRESULT SendDlgItemMessage (HWND  
    hDlg, int nIDDlgItem, UINT  
    Msg, WPARAM wParam, LPARAM  
    lParam) ;
```

Transferul datelor (2)

Parametrii funcției

- `hDlg` - *handle* către dialog
- `nIDDlgItem` - identificatorul controlului
- `Msg` - mesajul transmis
- `wParam, lParam` - parametrii mesajului

Exemple de mesaje - edit box

- WM_GETTEXTLENGTH
 - preluare lungime text
- WM_GETTEXT
 - preluare text
- WM_SETTEXT
 - modificare text

Exemple de mesaje - combo box

- CB_ADDSTRING
 - adăugare element
- CB_GETCOUNT
 - preluare număr elemente
- CB_GETLBTEXT
 - preluare element
- CB_GETCURRESEL
 - preluare index element selectat