

Facultatea Calculatoare, Informatica si
Microelectronica

Universitatea Tehnica a Moldovei

Medii Interactive de Dezvoltare a
Produselor Soft

Lucrarea de laborator#2

GUI Development

Autor:
Pastuh Vitalie TI 151

lector asistent:
Irina Cojanu
lector superior:
Radu Melnic

1 Scopul lucrarii de laborator

Realizarea unui simplu GUI Calculator

2 Obiective

1. Realizeaza un simplu GUI Calculator
2. Operatiile simple: +, -, *, /, putere, radical, Inversare Semn(+/-), operatii cu numere zecimale.
3. Divizare proiectului in doua module - Interfata grafica (Modul GUI) si Modulul de baza (Core Module).

3 Mersul lucrării de laborator

3.1 Cerintele

1. Realizeaza un simplu GUI calculator care suporta urmatoare functii: +, -, /, *, putere, radical, InversareSemn(+/-), operatii cu numere zecimale.
2. Divizare proiectului in doua module - Interfata gra ca (Modul GUI) si Mod-ulul de baza (Core Module).

3.2 Analiza Lucrării de laborator

Linkul la repozitoriu <https://github.com/TomaAna/MIDPS>

Sunt mai multe modalitati de realizare a unui gui calculator. Pentru crearea calculatorului am folosit programarea in windows cu ajutorul limbajului de programare C++.

Primul pas este crearea functiei WINMAIN care este eschivalentul WINDOWS a functiei main utilizata in toate programele scrise in C si C++, folosita pentru prelucrari primare. Functia WinMain difera insa in multe privinte de main si nu in ultimul rind, prin modul de declarare. int WINAPI WinMain(HINSTANCE hInst, HINSTANCE hPrev, LPSTR CmdLine, int CmdShow).

Functia WinMain returneaza o valoare int la fel ca multe alte programe in C++. Parametrii acceptati de WinMain:

```
WNDCLASS Wc;  
MSG Msg;
```

Functia WinMain din program termina cu o bucla while care preia mesaje pina cind utilizatorul trimite sistemului mesajul WMQUIT

```
Bucla While  
while(GetMessage(Msg, NULL, 0, 0))
```

```
HWND hActiveWindow = GetActiveWindow();  
if(!IsWindow(hActiveWindow) || !IsDialogMessage(hActiveWindow, Msg))
```

```
TranslateMessage(Msg);  
DispatchMessage(Msg);
```

```
return Msg.wParam;
```

Programul folosește clasa `BUTTON` pentru a crea butoane în cadrul ferestrelor.

Butoanele sunt butoane, de apăsare sunt în forma de dreptunghi.

```

HWND BCXButoane
(char Text; HWNDhWnd; intid; intX; intY; intW; intH; intStyle; intExstyle)
HWND A;
if(!Style)
Style=WSCHILDjWSV ISIBLEjBSM ULTILINEjBSP USHBUTTONjWST ABSTOP ;
if(Exstyle == 1)
Exstyle = WSEXSTATICEDGE; A = CreateWindowEx(Exstyle; "button"; Text; Style; X
BCXSscaleX; Y BCXSscaleY; W BCXSscaleX; H BCXSscaleY;
hWnd; (HMENU)id; BCXhInstance; NULL);
SendMessage(A; (UINT)WMSETFONT; (LPARAM)GetStockObject(DEFAULT_GUI_FONT);
(LPARAM)MAKELPARAM(FALSE; 0));
if(W == 0)
HDChdc = GetDC(A); SIZEsize; GetTextExtentPoint32(hdc; Text; strlen(Text); size); ReleaseDC(A; hdc);
return A;

```

```

HWND BCXEditeaza
(char Text; HWNDhWnd; intid; intX; intY; intW; intH; intStyle; intExstyle)
HWND A; ==assigndefaultstyleif(!Style)Style = WSCHILDjWSV ISIBLEjESW ANTRETU
SendMessage(A,(UINT)WMSETFONT; (LPARAM)GetStockObject(DEFAULT_GUI_FONT);
(LPARAM)MAKELPARAM(FALSE; 0));
return A;
char BCXGetText(HWNDhWnd)
inttmpint; tmpint = 2 + GetWindowTextLength(hWnd); char strtmp = BCXTmpStr(tmpint);
intBCXSetText(HWNDhWnd; char Text)
return SetWindowText(hWnd; Text);

```

4 Concluzie

În urma efectuării lucrării de laborator numărul 2 la MIDPS am studiat și am învățat cum să realizez un simplu GUI calculator care suporta următoarele

functii: +, -, /, *, putere, radical, InversareSemn(+/-), operatii cu numere zecimale. Am folosit un limbaj cu care am facut cunostinta recent, astfel am invatat si am analizat mai multe lucruri noi. Am utilizat un IDE pentru limbajele de programare C++, C ce a fost lansat in versiune stabila in 2008 care poarta denumirea de Code::Blocks care permite proiectarea interfetelor gra ce ntr-un mod vizual, de tipul WYSIWYG (What You See Is What You Get). Designerul se numeste wxSmith si este derivat din biblioteca wxWidgets, librarie ce permite crearea de interfete gra ce cross-platform.