

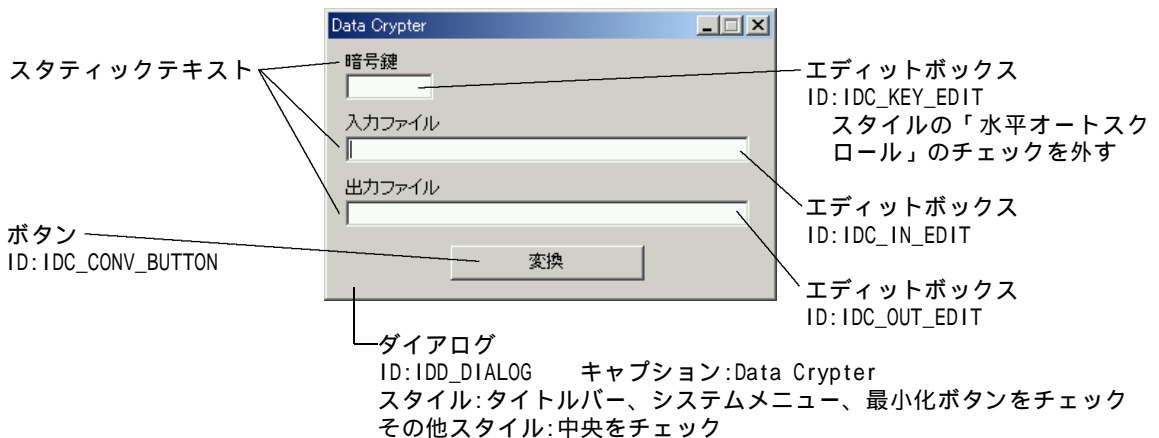
オブジェクト指向と ゲームプログラミング

基礎編 - 第18回 GUI

課題

暗号化および復号ツールをGUI化しましょう。

- 1 新規プロジェクトを「Win32 Application」で作成しましょう。
- 2 GUIとなるダイアログボックスを作成します。メニューから「挿入(I) リソース(R)」を選択します。「Dialog」を選択した状態で新規作成ボタンをクリックします。
- 3 以下のようなダイアログボックスを作成しましょう。



- 4 リソースを保存し、保存されたリソーススクリプトファイル(.rc)をプロジェクトに追加します。
- 5 以下のソースファイルを作成し、プロジェクトに追加しましょう。

- crypt.cpp -

```
/*
=====
                        ツールプログラミング
    Programmed by Hibikino software. Copyright (c) 2004 Hibikino software. All rights reserved.
=====
【対象OS】
    Microsoft Windows98/2000以降

【コンパイラ】
    Microsoft VisualC++ 6.0J ServicePack6

【プログラム】
    crypter.cpp          データ暗号化・復号

【履歴】
    * Version    1.00      2004/10/dd hh:mm:ss
=====
*/

/*
=====
                        インクルードファイル
=====
#include <windows.h>
#include "resource.h"

/*
=====
*/
```

```

/*                                     プロトタイプ                                     */
/*****
BOOL CALLBACK DialogProc(HWND hDlg, UINT uMsg, WPARAM wParam, LPARAM lParam);

/*****
/*                                     WinMain関数                                     */
/*****
int WINAPI WinMain(HINSTANCE hInstance, HINSTANCE hPrevInstance, LPSTR lpszCmdLine, int nShowCmd)
{
    return DialogBox(hInstance, MAKEINTRESOURCE(IDD_DIALOG), NULL, DialogProc);
}

/*****
/*                                     ダイアログボックスプロシージャ                                     */
/*****
BOOL CALLBACK DialogProc(HWND hDlg, UINT uMsg, WPARAM wParam, LPARAM lParam)
{
    return 0; // 処理しないメッセージはデフォルト処理
}

```

ダイアログボックスベースのアプリケーションなので、WinMain関数でDialogBox関数を呼び出してダイアログボックスを作成します。DialogBox関数は、ダイアログボックスの処理が終わるまで(EndDialog関数が実行されるまで)次の処理に進みません。また、DialogBox関数が内部的にメッセージループを確立するので、メッセージループを記述する必要はありません。

ダイアログボックスもウィンドウから派生したものであるため、ウィンドウプロシージャのようなメッセージを処理する関数(プロシージャ)が必要になります。その関数がDialogProc関数です。ダイアログボックスでなにかイベントが起こると、この関数が呼び出されます。ダイアログボックスのプロシージャでメッセージを処理した場合は0以外の値、処理しない場合は0を返します。0を返した場合、システムによりそのメッセージが適切に処理されるので、DefWindowProc関数を呼び出す必要はありません。