

オブジェクト指向と ゲームプログラミング

基礎編 - 第19回 ダイアログプロシージャ

課 題

ダイアログプロシージャを作成し、メッセージが適切に処理されるようにしましょう。

1 ダイアログボックスを終了できるようにしましょう。

第7回の状態では、ダイアログボックスの閉じるボタンを押してもダイアログは消えず、終了できません。ダイアログボックスを終了させるには、ある関数を呼び出します。閉じるボタンが押されると、あるメッセージが発生します。このメッセージが発生したタイミングである関数を呼び出します。

以下のプログラムをダイアログプロシージャの適切な場所に追加しましょう。

```
// 閉じるボタン
case WM_?????:
    ???????? (hDlg, 0); // ダイアログボックスの終了
    return 1;
```

2 システムメニューにオリジナルのアイコンを表示しましょう。

第7回の状態では、システムメニューのアイコンがありません。以下の手順に従ってアイコンを作成し、システムメニューに表示しましょう。

- (1)メニューから「挿入(I) リース(R)」を選択します。「Icon」を選択した状態で新規作成ボタンをクリックします。
- (2)「ステール (16x16)」と「32x32, 256色」のアイコンを作成します。システムメニューは、「ステール (16x16)」のアイコンが使用されます。
- (3)ダイアログボックスが作成されたときにアイコンを読み込むようにプログラムを変更します。

ダイアログボックスが作成されると、あるメッセージが発生します。このメッセージが発生したタイミングで、アイコンを読み込んでダイアログボックスに関連づけます。

以下のプログラムをダイアログプロシージャの適切な場所に追加しましょう。

```
// ダイアログ初期化
case WM_?????????:
    // アイコン設定
    hIcon = ???????? (GetModuleHandle(NULL), ?????????????????? (IDI_ICON1)); // アイコンの読み込み
    SetClassLong (hDlg, GCL_?????, (LONG)hIcon); // アイコンの変更

    return 1;
```

3 変換ボタンが押されたときの処理を作成しましょう。

ダイアログボックスのボタンが押されると、WM_COMMANDメッセージが発生します。このとき、wParamの下位16ビットには、押されたボタンのIDが格納されます。

押されたボタンが変換ボタンだった場合には、入力ファイルと出力ファイルの名前をGetDlgItemText関数でエディットから取得し、それぞれのファイルを開きます。入力ファイルから1バイトずつ読み込み、それを暗号化または復号して出力ファイルに書き出します。

以上をまとめると、以下のようなプログラムになります。ダイアログプロシージャの適切な場所に追加しましょう。

```
// ボタン
case WM_?????:
    switch( wParamの下位16ビットを取り出す処理をここに記述します ) {
        // 変換ボタン
```

```

case IDC_   ここは各自考えましょう   :
// 暗号鍵をエディットから取得
int   nKey;
nKey = GetDlgItemInt(hDlg, IDC_   ここは各自考えましょう   , NULL, FALSE);

TCHAR   szFileName[MAX_PATH + 1];

// 入力ファイル名をエディットから取得
if(GetDlgItemText(hDlg, IDC_   ここは各自考えましょう   , szFileName, MAX_PATH) == 0) {
    ??????????(hDlg, "入力ファイルが未入力です", "エラー", MB_ICONINFORMATION | MB_OK);
    break;
}

// 入力ファイルを開く
FILE*   fpInput = ?????(szFileName, "rb");
if(fpInput == NULL) {
    ??????????(hDlg, "入力ファイルが開けません", "エラー", MB_ICONINFORMATION | MB_OK);
    break;
}

// 出力ファイル名をエディットから取得
if(   ここは各自考えましょう == 0) {
    ??????????(hDlg, "出力ファイルが未入力です", "エラー", MB_ICONINFORMATION | MB_OK);
    break;
}

// 出力ファイルを開く
FILE*   fpOutput = ?????(szFileName, "wb");
if(fpOutput == NULL) {
    ??????????(hDlg, "出力ファイルが開けません", "エラー", MB_ICONINFORMATION | MB_OK);
    fclose(fpInput);
    break;
}

// 暗号鍵で乱数初期化
srand(nKey);

// 暗号化および復号
char   c;
while(fread(&c, sizeof(char), 1, fpInput) != 0) {
    const int   r = rand() / 256;
    c ^= r;
    fwrite(&c, sizeof(char), 1, fpOutput);
}

// ファイルを閉じる
??????(fpOutput);
??????(fpInput);

????????(hDlg, "変換終了", "終了", MB_ICONINFORMATION | MB_OK);

break;
}

return 1;

```