

Visual C++利用法入門

鈴木, 孝彦
九州大学情報基盤センター

<https://doi.org/10.15017/1470505>

出版情報 : 九州大学情報基盤センター広報 : 学内共同利用版. 2 (3), pp.147-159, 2002-11. 九州大学
情報基盤センター
バージョン :
権利関係 :

Visual C++ 利用法入門

鈴木 孝彦¹

本学の教育用システムの PC 上では、Microsoft 製の C++ プログラム開発環境「Microsoft Visual C++ (R) 6.0」が利用できます²。この記事では、C++ を使って Windows 上の高性能なアプリケーションを作ってみたいけれど使い方がわからない、なんだか知らないけど面白そうだから Visual C++ を使ってみたい、といった方を対象に、Visual C++ 上で簡単なプログラムを動作させる方法を紹介します。

1. はじめてプログラムを入力して実行する

```
#include <iostream>
using namespace std;

int main()
{
    cout << "Hello World!" << endl;
    return 0;
}
```

図 1 入力するプログラムテキスト

図 1 は、「Hello World!」という文字列を出力するための、簡単な C++ プログラムです。このプログラムを Visual C++ を使って入力、実行してみましょう。手順は次のようになります。

Visual C++ 開発環境の起動

まず、Windows のスタートボタンから「プログラム」→「Microsoft Visual Studio 6.0」→「Microsoft Visual C++ 6.0」を選択して、Visual C++ アプリケーションを立ち上げます。Visual C++ では、プログラムテキストの入力、チェック、実行、その他のプログラム開発に必要な作業が行えます。Visual C++ のようなアプリケーションをプログラムの“開発環境”と呼びます。

¹ 情報基盤センター E-mail:suzuki@cc.kyushu-u.ac.jp

² 最新版の Visual C++ .NET でも基本的な操作は共通しています。

プロジェクトの新規作成

Visual C++ では作成したいプログラム各々について、名前をつけた“プロジェクト”を作ります。プログラムに必要な各種の設定をプロジェクトに登録することが、Visual C++ でのプログラム作成作業です。**プログラムを作るためにはプロジェクトを作る**、と覚えてください³。

Visual C++ のメニューから「ファイル」→「新規作成」を選択します。すると図 2 のようなダイアログが現れて“新規作成”の選択画面となります。

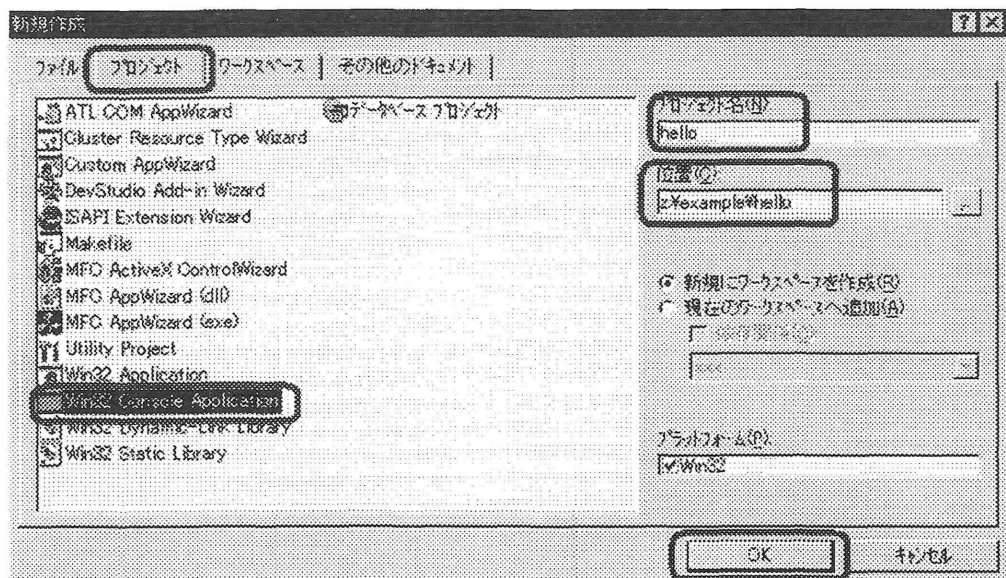


図 2 プロジェクト新規作成選択画面

ここで、「プロジェクトタブ」を選択し、「Win32 Console Application」を選択します。さらに「プロジェクト名」と「位置」を入力します。ここでは「プロジェクト名」を hello 「位置」を Z:\example\hello と指定しました。プロジェクトは、「位置」で指定したディスク上のフォルダに保存されます。

このプロジェクトは“コンソールアプリケーション”，つまり、文字が表示出来る画面とキーボードだけを使うプログラムです。C++ 言語の入門書に掲載されている初歩的なプログラムの多くはコンソールアプリケーションです。

「OK」ボタンを押すと、図 3 に示すようなウィンドウが表示されますので、「空のプロジェクト」を選択し、「終了」ボタンを押します。Visual C++ がさらにもう一つウィンドウを表示しますのでもう一度「OK」ボタンを押してください。

³ プロジェクトを複数含むワークスペースというのがありますが、省略します。

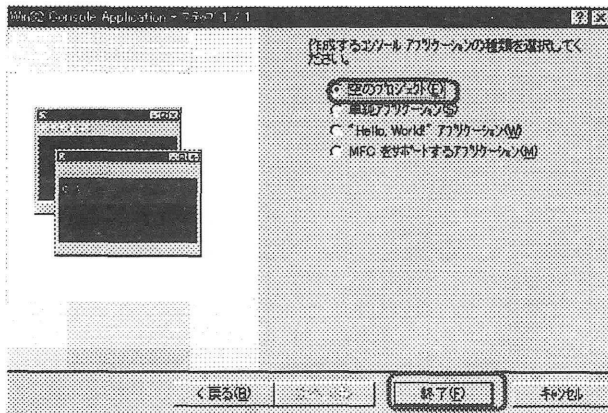


図 3 空のプロジェクト選択画面

ソースファイルの新規作成

図 1 のプログラムを「hello」プロジェクトに登録します。Visual C++ のメニューから「プロジェクト」→「プロジェクトへ追加」→「新規作成」を選択します。すると図 4 のようなダイアログが現れますので、「ファイルタブ」を選択、続いて「C++ ソースファイル」を選択し、「ファイル名」と「位置」とを入力して「OK」ボタンを押します。ファイル名は `hello.cpp` 位置は `z:\example\hello` とします。

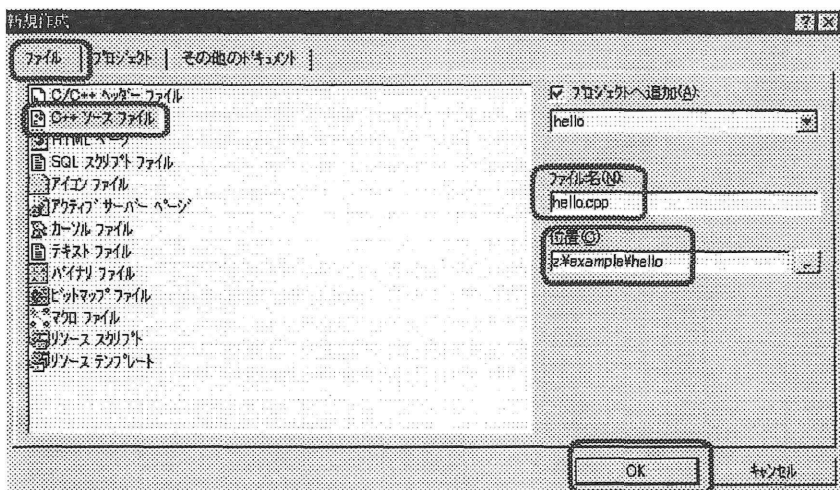


図 4 ソースファイルの新規作成

「OK」ボタンを押すとテキストの入力が可能になりますので、図 1 のプログラムをキーボードから入力します。これでプログラムテキスト（ソースファイル）が登録されました。

ビルド

プロジェクトに登録したソースファイルから、実行可能なプログラムを作成します。Visual C++ メニューから「ビルド」→「ビルド」を選択します。ここまでの作業を正しく行っていれば、しばらくして Visual Studio のウィンドウ下部に“hello.exe · エー 0、警告 0”と表示されるはずです。

実行

「ビルド」「実行」を選択すると、コンソール画面（通常の設定では 80 桁 x25 行の黒い背景の画面）が現れて

```

Hello World!
Press any key to continue
    
```

と表示されます。表示されたら Enter キーを押してコンソール画面を終了させましょう。

ビルドがうまく行かない場合の対応

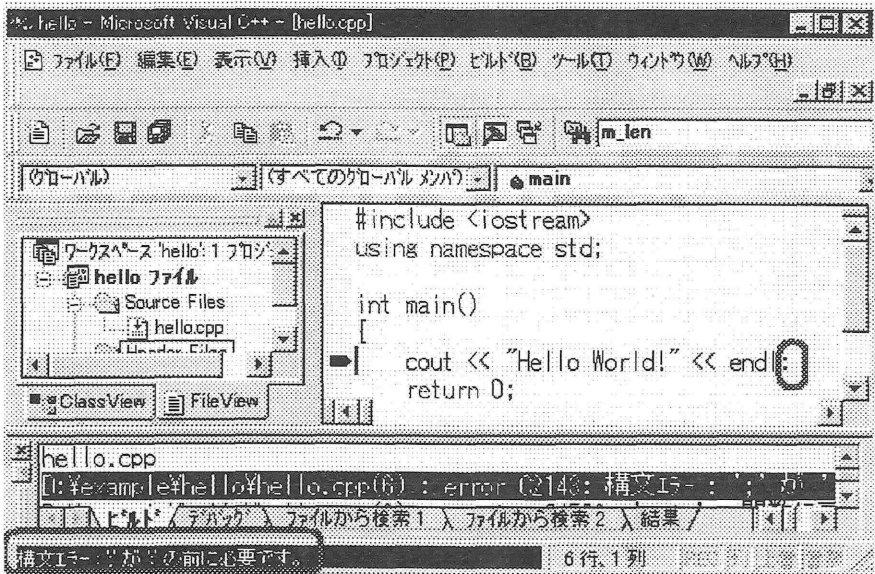


図 5 ビルドでエラーが発生した場合

ビルドを実行した際、“エラー 0、警告 0”と表示されずに、エラーの数が表示されたときは、(キーボードの) F4 キーを押します。すると、カーソルがエラーのある入力テキスト付

近に移動します。図5の例では、セミコロン「;」を入力すべき場所に、誤ってコロン「:」を入力していました。テキストを修正してもう一度ビルドしましょう。この他に、カッコの対応、二重引用符「"」の対応、つづり間違い、漢字変換モードでの入力などに注意が必要です。一個所の入力ミスでエラーが大量に発生することもあります。

すべてをチェックしてもわからない場合は、Visual C++ 開発環境を一旦終了し、新しいプロジェクト hello2 を作ってやり直してみましょう。プロジェクトの種類が「空のプロジェクト」になっていなかったり、入力したプログラムテキストに誤って 2 バイト文字が混入していることがあります。

2. ウィンドウを表示するプログラムを作成する

次に、ウィンドウを表示するプログラムを作ってみましょう。手順は少し面倒ですが、難しいことはありません。ウィンドウを表示する基本的な機能は Visual C++ が自動的に用意します。そのウィンドウの上に“コントロール”と呼ばれる部品（ボタンやテキスト）を追加登録し、それぞれのコントロールに対応するプログラムコードを記述します。

プロジェクトの新規作成

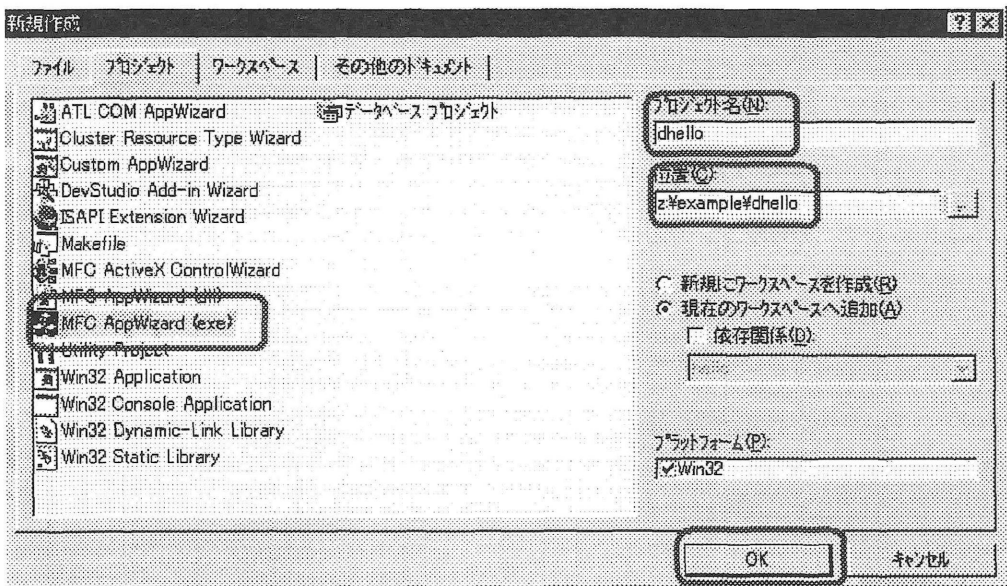


図 6 dhello プロジェクトの新規作成

コンソールアプリケーションと同じようにプロジェクトを生成します。ただし、プロジェクトの種類として「MFC AppWizard (exe)」を選択し、プロジェクト名として dhello を

入力します。位置は `z:\example\dhello` とします (図 6)。「OK」ボタンを押すと、作成するアプリケーションの種類の選択画面になります。ここで、「ダイアログベース」に選択を変更し、「終了」ボタンを押します (図 7)。

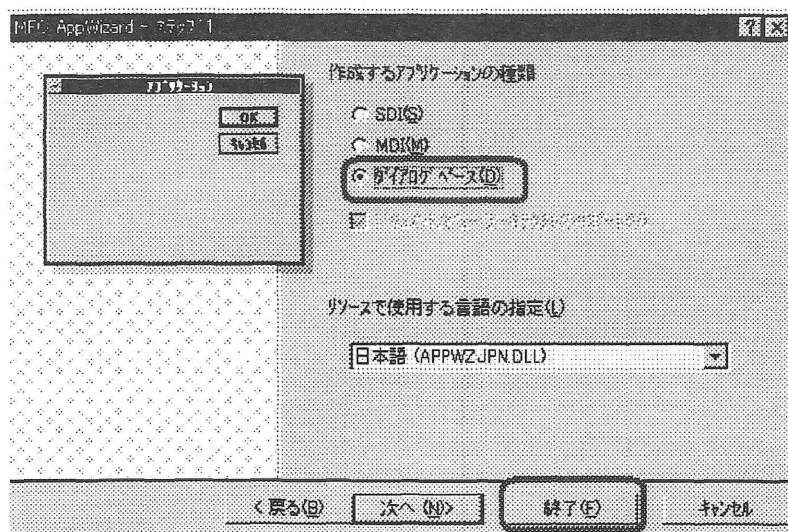


図 7 ダイアログベースアプリケーションの選択

ダイアログは、メニューも特別な描画画面も持たないウィンドウです。図 2,3,4,6,7 はダイアログの例です。なお、図 5 (Visual C++ 開発環境本体) のようなウィンドウは MDI (Multi Document Interface) といいます。

テストビルドと実行

MFC AppWizard のプロジェクトでは、Visual Studio がウィンドウを表示するために必要な最小限のプログラムコードを自動的に生成します。そのため、既にこの段階で、Visual C++ のメニュー「ビルド」「実行」を選択すると、プログラムが実行され図 8 に示すようなウィンドウ (ダイアログ) が表示されます。

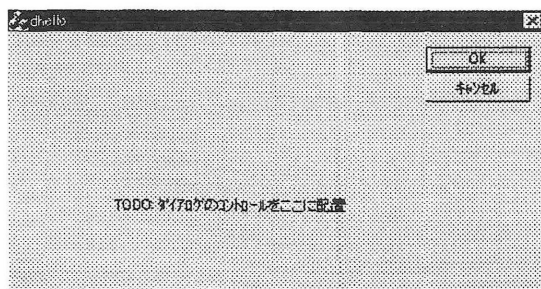


図 8 ウィンドウを表示するプログラムの実行テスト

ウィンドウの「OK」ボタンまたは「キャンセル」ボタンを押すとプログラムは終了します。

プログラムの変更

次に、プログラムに変更を加えます。一つめのボタンを押すと Hello World! と表示し、もう一つのボタンを押すと “Hello World!” の文字数(12文字)を表示する図9のようなウィンドウを作ってみましょう。

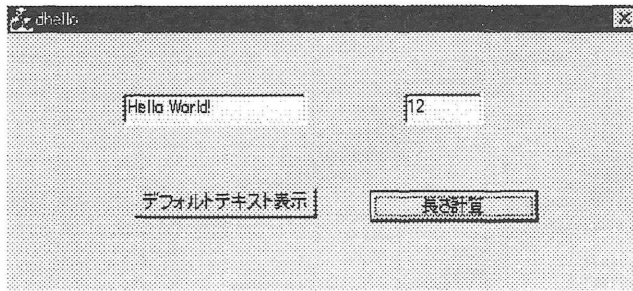


図 9 これから作成するプログラムのウィンドウの様子

ダイアログエディタの実行

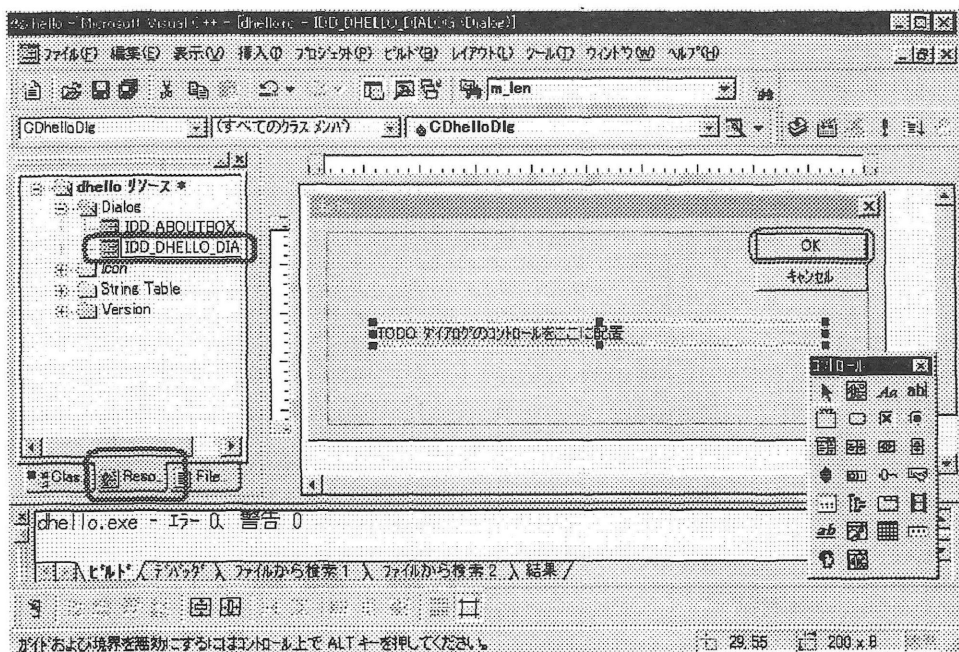


図 10 ダイアログエディタ

テストプログラムを終了させると、Visual Studio のウィンドウは、図 10 のようになります。こうならない場合は、画面左下の「リソースタブ」を選択し、「Dialog」アイコンを展開、さらに「IDD_DHELLO_DIALOG」をクリックしてください。図 10 の画面はウィンドウ上のコントロール（ボタンやメニュー）を編集する画面（ダイアログエディタ）です。

デフォルトコントロールの削除

ダイアログエディタの編集画面に表示されているウィンドウの「OK」ボタンをマウスでクリックして選択し、キーボードの DELETE キーを押すと、ウィンドウから「OK」ボタンが削除されます。同様に、「キャンセル」ボタンおよび“//TODO ...” テキストも削除します。

コントロールの追加

コントロールツールバー（図 11）で“ボタン”を選択し、ダイアログエディタ上のウィンドウのボタンを配置したい場所をクリックします。するとウィンドウ上にボタンが追加されます。

ボタンをもう一つと“エディットボックス”を2つ追加します。エディットボックスとは、テキストの入力、表示を行うための部品です。なお、設定によっては、コントロールツールバーの外見が図 11 と異なる場合がありますので注意してください。この段階で、ダイアログエディタの編集画面は図 12 のようになります。

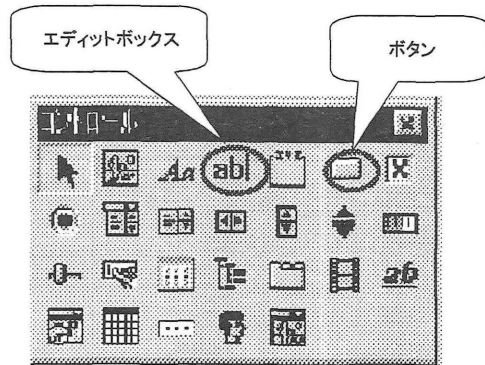


図 11 コントロールツールバー

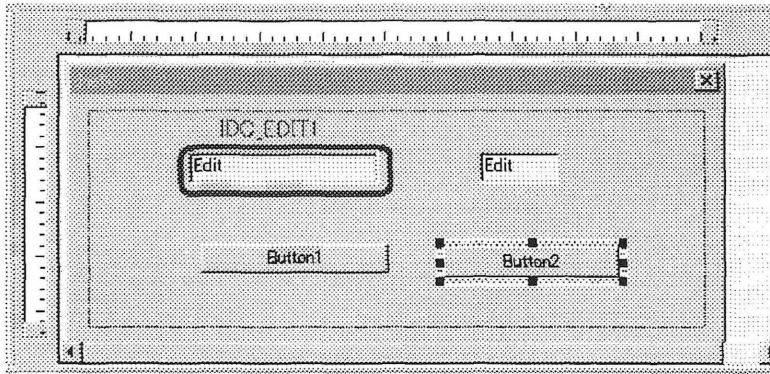


図 12 ダイアログエディタでのボタンとエディットボックスの追加

入力用変数の追加

ダイアログエディタの編集画面の上部左側に表示されているエディットボックス（図 12 の IDC_EDIT1）をマウスで選択後、右クリックし表示されるメニューから ClassWizard を選択します（図 13）。IDC_EDIT1 は、ウィンドウ上のコントロール（ボタンやテキスト入力部）それぞれを識別するための名前（コントロール ID）です。「コントロール ID」で、IDC_EDIT1 が選択されていることを確認した後「メンバ変数タブ」を選び「変数の追加」ボタンを押します。変数名を `m_hello` と入力し、「OK」ボタンを押して終了します。この変数(`m_hello`)は、エディットボックスに入力、表示されるテキストを記憶するために使われます。

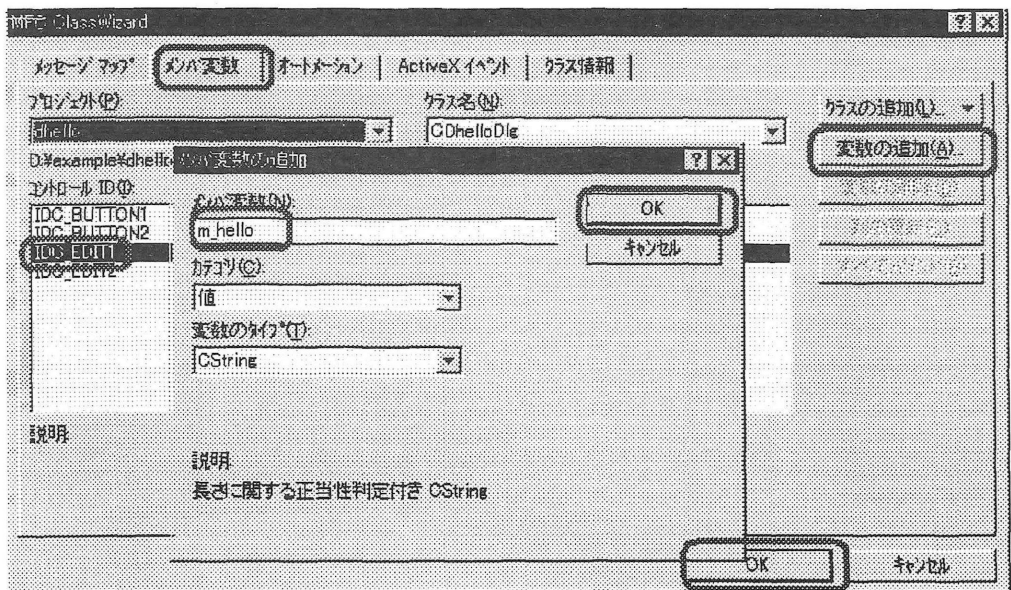


図 13 ClassWizard でのエディットボックスへの入力用変数追加

出力用変数の追加

同じく、右側のエディットボックス (IDC_EDIT2) に変数を追加します。変数名を `m_length` と設定します。設定画面は図 13 と同様です。ただし今度は、コントロール ID として、IDC_EDIT2 が選択されているはずで

デフォルトテキスト表示ボタンの設定

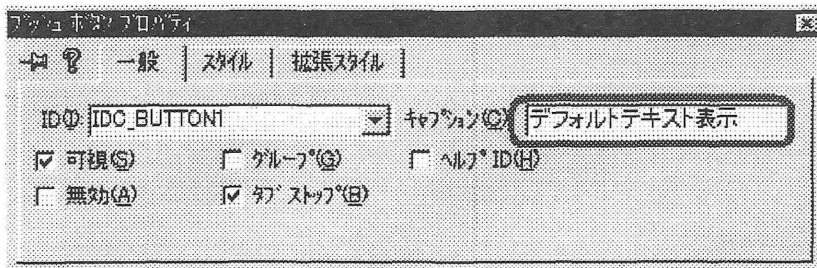


図 14 デフォルトテキスト表示ボタンのキャプション設定

ダイアログエディタ上で Button1 ボタンをマウスで選択し、右クリックして表示されるメニューから「プロパティ」を選択します。「キャプション」を「Button1」から「デフォルトテキスト表示」に変更します (図 14)。

デフォルトテキスト表示動作の設定

「デフォルトテキスト表示」ボタンをマウスで選択し (左) クリックして出てきたダイアログの「OK」ボタンを押すと、ボタンを押したときの動作を C++ コードで指定できます。入力内容は、図 15 の枠で囲まれた部分です。`m_hello` は、図 13 で設定した、エディットボックス IDC_EDIT1 に表示される入力テキスト用の変数です。`UpdateData(FALSE)` によって、`m_hello` の内容がエディットボックスに表示されます。

長さ計算ボタンと長さ計算動作の設定

ダイアログエディタに戻って (図 10 参照)、Button2 についても同様に、プロパティの「キャプション」を「長さ計算」に変更します (指定方法は図 13 参照)。「長さ計算」ボタンを押したときの動作は、図 16 のように指定します (入力方法は図 15 参照)。

図 16 では、1 行目の `UpdateData(TRUE)` で、エディットボックスに表示されているテキストを `m_hello` 変数に読み込みます。2 行目では `m_hello` テキストの長さを調べて (`GetLength`) 文字列に変換し (`Format`) `m_length` に設定します。3 行目で `m_length` の内容がエディットボックスに表示されます。

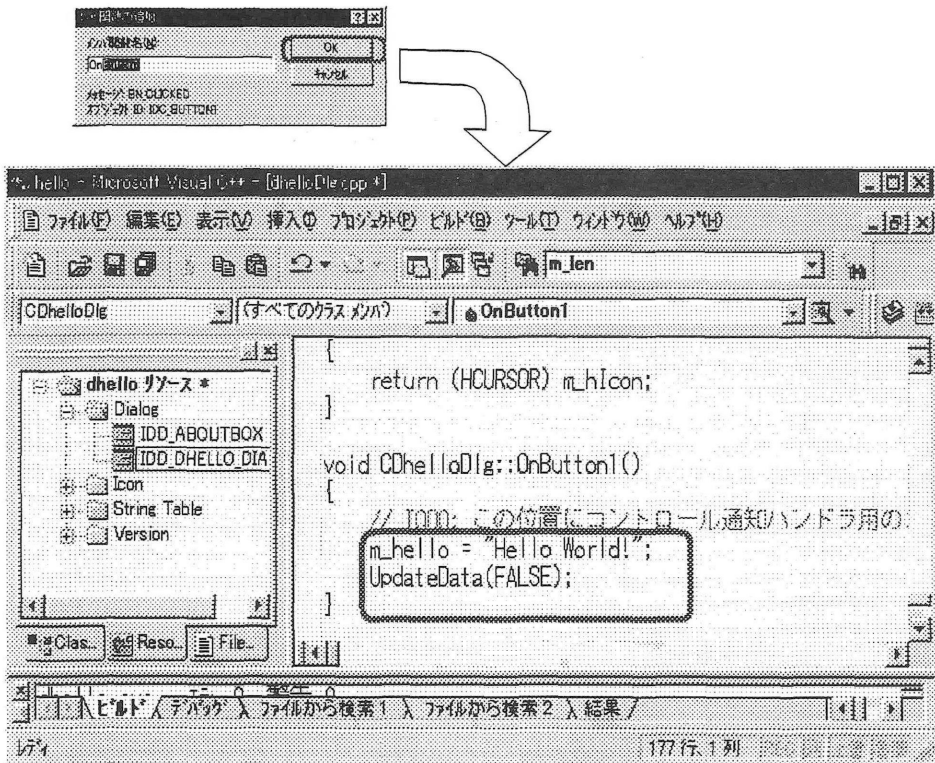


図 15 「デフォルトテキスト表示」ボタンを押したときの動作指定

```

UpdateData(TRUE);
m_length.Format("%2d", m_hello.GetLength());
UpdateData(FALSE);

```

図 16 長さ計算動作の設定コード

ビルドと実行

Visual Studio のメニューから「ビルド」「実行」を選択すると、作成したプログラムを実行しウィンドウが表示されます。「デフォルトテキスト表示」ボタンを押すと、左側のエディットボックスに Hello World! と表示されます。「長さ計算」ボタンを押すと、右側のエディットボックスに テキスト “Hello World!” の文字数、すなわち 12 が表示されます。

左側のエディットボックスのテキストを適当に書き換えてから「長さ計算」ボタンを押

すと、その時表示されているテキストの文字数が表示されます。ウィンドウ右上の[x]をクリックすると、プログラムが終了します。図 17 に実行の様子を示します。

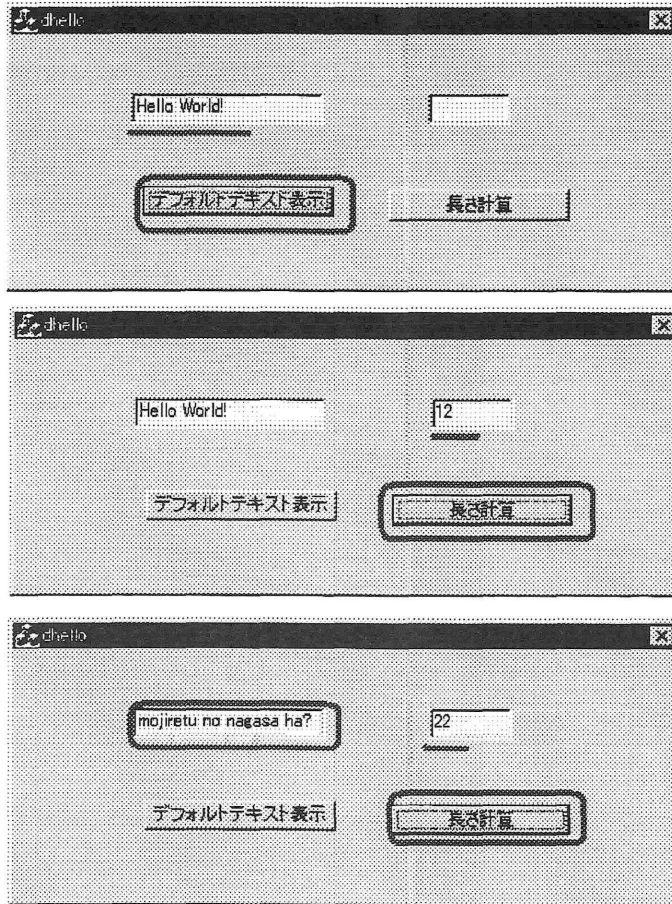


図 17 プログラム実行画面

3. 次に何をすれば良いか

ここまでで、Visual C++ を使って、コンソールアプリケーションと簡単なウィンドウを表示するプログラムの作成手順を示しました。それでは、自分の思う通りの Windows 上のプログラムを作れるようになるためには、これからどうしたらいいでしょう。大きく分けて、

- ・一般的なプログラミングの方法についての知識
- ・C++ の文法についての知識
- ・Visual C++ と Windows の機能についての知識

の三つが必要になります。それぞれについて、多くの入門書が出版されていますので、それらを購入するのも良い方法です⁴。ここではもう少し手軽に出来る方法を紹介します。

Web で検索する

Web 上の検索エンジン (<http://www.google.co.jp> 等) を使って、Web 上の資料を探してみましょう。検索キーワードとして {“C++”, “講座”}, {“Visual C++”, “入門”} などで検索すると、きっと役に立つ Web ページが見つかるはずです。

Web ページは、書籍の入門書に比べて作成者の好みが反映された内容になりがちです。自分の必要とする内容が書いてあるページを探しましょう。なお、この記事を書くに当たって、Web ページ http://www.nara-edu.ac.jp/~asait/visual_cpp/intro_cpp.htm が参考になりました。

オンラインヘルプを使う

Visual Studio には、大量のオンラインヘルプ情報 (MSDN ライブラリ) が付属しています。たいいていの情報はオンラインヘルプの中に記述されています。ただし、あまりにも情報が大量であるため、一から探そうとしても肝心な情報にたどり着くまでに疲れてしまう場合があります。Web 上で調べた内容の確認、補完として使うのがよいでしょう。

少し腰をすえて Windows 上の Visual C++ プログラミングを勉強したい人は、Visual C++ オンラインヘルプ上で、“scribble チュートリアル”を検索して参考にしてください。

App Wizard の生成したプログラムを参考にする

AppWizard(exe) は、自動的にウィンドウを表示するプログラムを生成します。自動生成されたプログラムがどういう構造になっているか、ボタンや他のコントロールを追加したときに、プログラムにどのようなコードが追加されるのかを確認しておきましょう。理解の助けになるはずです。

⁴ “C++プログラミング入門”，グレゴリー サティア他著，オライリー・ジャパン，ISBN: 4873110637。“標準講座 MFC6.0—Visual C++による効率的な Windows プログラミング Programmer’s SELECTION”，ハーバート シルト著，1999，翔泳社，ISBN: 4881357042 等。