

NUnitを使った

ハイクオリティ プログラミングの ススメ

新連載

テストフレームワークによる.NET開発

浅井 斉 *Asai, Hitoshi*
株式会社テクノロジックアート
テクニカルデプトシステム
デベロップメントグループ

テストフレームワークを知る

Technology Tools

- Visual Basic .NET
- Visual C# .NET
- SQL Server 2000
- Oracle 9i
- Access 2002
- ASP.NET
- Internet Information Services
- Other:
NUnit

Level

Samples

・この記事で取り上げたソースコードおよびサンプルプログラムは、付録CD-ROMの¥DOTNET¥HIQUALITYディレクトリに収録しています。

¥CALCULATOR.NET
VB.NET版サンプル

¥CALCULATOR.VB
VB版サンプル

*) 記事内では日本語のプロジェクト名などを使用していますが、収録したサンプルは英字を使用しています。

はじめに

「テストテスト 楽しいなっ」なんて経験をお持ちでしょうか？ これまで、テストはとても苦痛を伴う作業だったかと思います。何度も何度も手作業で同じことを繰り返して繰り返して……。でも、少しだけ考え方を变えて、プログラミングのやり方を変えると、テストは変わります。どう変わるかって？ そりゃもう、たまたま楽しくいものに、ね。

連載第1回目の今回は、テストフレームワークとは何かを知るために、従来のテスト手法にテストフレームワークを適用してゆくところから始めます。テストフレームワークの本質は、テスト駆動開発のような従来と異なる考え方に適応するものですが、従来のままの考え方に適用するだけでも、より便利で効率の良いテス

ト/デバッグ手法を体験することができます。

昔ながらのテスト/ デバッグ手法

テストの復習

さて、あなたはVisual Basicを用いて開発を行なう際に、どのようなテスト/デバッグ手法を使ってきました？

仮に、あなたが大規模なソフトウェア開発のチームにいて、管理された「プロセス」があった場合は、特殊なテスト用ツールを使うか、あるいはテストのための丁寧な手順をこなしてきたはず。逆に、小規模なソフトウェア開発の場合、あるいは趣味でプログラミングをするような場合には、ツールも手順も関係なく、テスト/デバッグの手法など、ほとんど存在しないに等しい状態だったのではないのでしょうか。

図1：計算プログラムの画面



もちろん、後者のように特別なツールや丁寧な手順を用いない場合でも、作成したコードが正しいかどうかは確認しなければなりません。そのために、Visual Basicの機能を利用して、以下のような手順で動作確認をしておいたはずで

- 手順- : [F5] キーを押して、書いたコードを実行
- 手順- : 表示されたフォームを操作する
- 手順- : 操作の結果を目で見て確認する

このとき手順- で確認した結果が、頭の中で想定していた結果と同じであればコードは正しいこととなります。そうでなければコードが間違っていることとなります。この手順ののっとり、想定していた機能が正しく動作する/しないを確認することはできるわけです。

たとえば、図1のような画面で、次に示すような機能を持った「計算プログラム」を、Visual Basicで作成し

リスト1 計算プログラムのソースコード

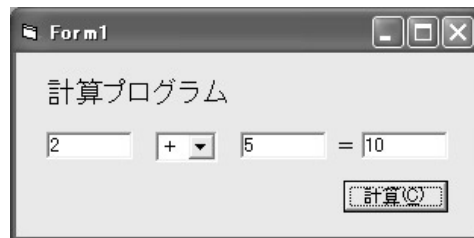
```
Private Sub Command1_Click()
    Dim leftOperand As Double
    Dim rightOperand As Double
    Dim operatorType As String
    Dim resultValue As Double

    leftOperand = Text1.Text
    rightOperand = Text2.Text
    operatorType = Combo1.Text

    resultValue = _
        Calculate(leftOperand, rightOperand, operatorType)

    Text3.Text = resultValue
End Sub
```

図2：実行結果の確認



たとしましょう(リスト1)

- ・2つのテキストボックスに数値を入力できる
- ・演算子を選択できる
- ・[計算] ボタンを押したら、数値と演算子に基づいて結果を表示する

先ほどの手順を利用して、このプログラムが正しいかどうかを確認するためには、まず[F5]キーを押して、プログラムを実行します。

次に、表示されたフォームに対してプログラムの機能を「2 + 5 = 7」や「15 - 7 = 8」などのように想定し、実際にそのフォームを操作します。そして、その操作の結果を目で見て確認し(図2)、その結果が、頭の中で想定していた結果と同じであれば、プログラムは正しく書けていたこととなります。

このような、実際にコードを実行し動作を確認するこ

```
Private Function Calculate(leftOperand As Double, _
    rightOperand As Double, _
    operatorType As String)
    Dim resultValue As Double

    If (operatorType = "+") Then
        resultValue = leftOperand * rightOperand
    ElseIf (operatorType = "-") Then
        resultValue = leftOperand - rightOperand
    ElseIf (operatorType = "x") Then
        resultValue = leftOperand * rightOperand
    ElseIf (operatorType = "÷") Then
        resultValue = leftOperand / rightOperand
    End If

    Calculate = resultValue
End Function
```

*) このサンプルコードは、デバッグを行なうために間違いを残してあります。