Jan Ruchel[*]

Creating Applications for Geodetic Computations[**]

1. Visual Basic – Simple and Effective Programming Language

The computations related to geodetic works require an application guaranteeing execution of computations. Therefore, a program performing the required computations can be purchased, or an own application for execution of computations can be created, including specifying details of realization of a task in a complete way as well as the method and the range of presentation of the results.

An own program can realize problems from the simplest computations to complicated analyses. We decide which functions are executed by our application.

The commercial programs that are nowadays available at the market offer final results and often do not provide any view into indirect results (that are repeatedly required for making an analysis, particularly if any computation troubles occur). If an own program is created, it is possible to make a view into indirect results at any computation stage.

An application performing computations that are interesting for us, can be created in many ways:

- write a program from the ground up taking advantage of the existing market programming tools (e.g. C++, Delphi, Java, Visual Basic or another);
- take advantage of the popular Excel spreadsheet; in addition to broad calculation capabilities (including the matrix algebra), the Excel spreadsheet features embedded elements for programming the macroinstructions by means of the Visual Basic language; thereby, a complete application for a given group of computation problems can be created.

[*] Faculty of Mining Surveying and Environmental Engineering, AGH University of Science and Technology, Krakow

[**] Realized within the scope of granted researches no. 11.11.150.006

The example of employment of the Visual Basic programming language for creation of a computing application (related to adjustment of a levelling net) is presented below.

During the first years of development of the computer science, the programmers were regarded as the extremely talented persons, and programming as inaccessible for common people. The programming languages including low-level (assembler) and high-level languages (mainly C++) developed together with the hardware. Nowadays, they are less complicated and feature many engines that make programming easier and are more available for common computer users.

The Basic language was created in 1963 and was a very simple programming language designed for beginning programmers. A break-through occurred when Visual Basic 1.0 was implemented. It was developed along with introduction of the Microsoft Windows operation system. Visual Basic was already a visual language and many programming procedures were executed through graphical arrangement of objects on the application window (form). It was not required to write very complex source codes which most elements of the programmer work were used for creation of an application form. As for Visual Basic it was possible for a programmer to create quickly a fully functional application taking advantage of the graphical user interface (IDE). Subsequent versions of Visual Basic featured more and more tools simplifying program development. All versions were backward compatible, i.e. they had all advantages of earlier versions (as well as faults). The next development step was introduction of VB.NET version in 2002. The programs were not compatible anymore due to the changes of the language syntax (this version is even called a new programming language). VB.NET guarantees that the elements executed in many programming languages can be consolidated within a single design and the applications operating on various system platforms can be created (as for more advanced programmers).

2. Creation of an Event-Type Application (Program Form and Code)

An application written in Visual Basic consists of two elements:

1) Graphical user interface, i.e. the system of the window (called also as the form) and the objects assigned to the window. It is possible to take advantage of many linked windows (including a start window) within a complex design.
2) Program code containing procedures together with the set of instructions to be executed.

The applications written in Visual Basic are event-type programs, i.e. a correct realization procedure is triggered if an appropriate event occurrs. An event can be any of the following activities performed by a user: clicking a mouse button, pressing a proper key, marking by a cursor, etc. The events can be also linked with the external devices (e.g. any measuring instrument) or be the effect of passing time. Occurrence or non-occurrence of an appropriate event conditions the order of an executed program code. Therefore, it is necessary to guarantee a proper order of events during programming. It can be achieved through controlling either activity (or non-activity) or availability of each object. If an application is created, the following rule shall be generally observed: do not empower a user to make a mistake, otherwise a user makes it for sure.

An event shall be linked with a particular user interface object within our application, if it is to trigger an appropriate reaction as the executed instructions.

The event service procedure has the following syntax:

```
Private Sub Object_event (optional procedure parameters)
....
'list of instructions assigned to an object and an event
.....
End Sub
```

As it is shown above, the name of the procedure contains the name of an object and an event associated with the object. It can also contain appropriate parameters linked with the type of an object and the type of an event. The headers of the procedures are automatically generated by Visual Basic, e.g. Private Sub Button_Click (ByVal sender As Object, ByVal e As System.EventArgs) Handles button.Click.

The program features a very complex (consisting of many windows) and user-friendly interface that makes creation of applications an effortless process.

The most important windows are as follows:

- Solution Explorer containing all elements of our design. The elements are appropriate directories and files generated automatically by Visual Basic, as well as forms and modules added during elaboration of a design by a programmer.
- View Designer – an application can be designed in its graphical mode.
- Toolbox containing (appropriately grouped) lists of available objects used in the interface design mode.
- Properties showing current values of all features for an active object in the design window. The values of selected features for a pointed object can be modified in this window. If necessary, the features can be also changed at the program code level during execution of each procedure (Fig. 1).

- View Code – used for writing a program code consisting of the procedures assigned to each object and their associated events as well as any declaration and other functions and procedures (not associated with particular objects) (Fig. 2).
- Many other windows making it possible to develop and control complex designs.

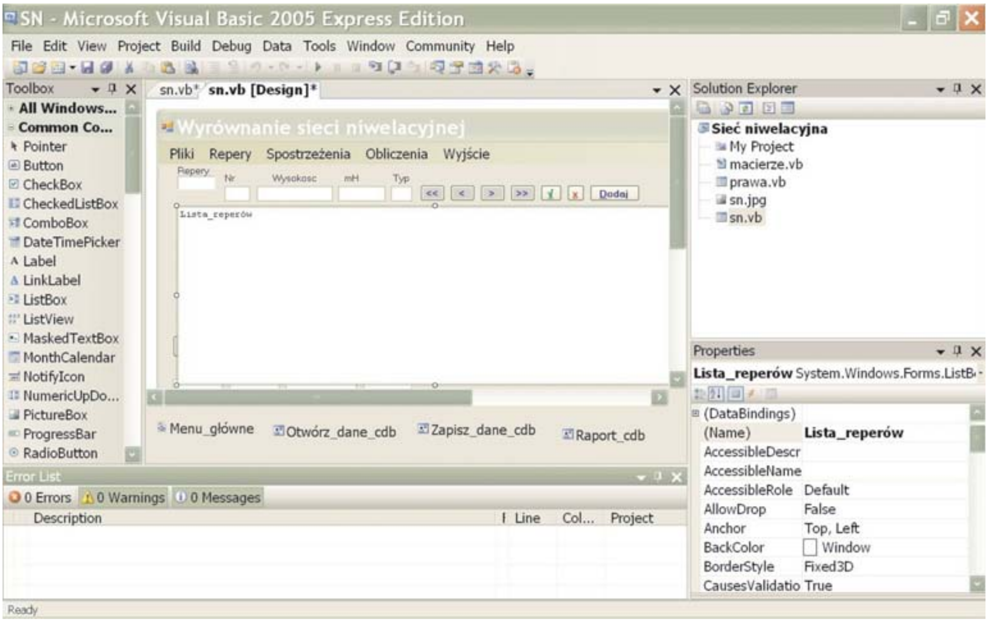
In order to add objects to our design, the objects shall be put on the form by means of a mouse cursor, and the form size and position shall be specified by means of a mouse and actions related to the Windows graphical elements. We fix the value of each feature for a given object using the Properties window.

Visual Basic is a user-friendly programming language due to the embedded error control. The errors occurring during writing a program can be separated into a few types:

- Errors during text edition – they can be avoided through a careful control of the writing process; these errors are not critical for an application.
- Errors discovered during compilation – the errors that prevent compilation of a program, such as syntax mismatches for declarations or instructions, or errors of program element identification.
- Program execution errors – occurring during execution of an algorithm element; in order to prevent them, the proper securities, i.e. error traps, are applied during writing a program code.
- Logical errors – the most critical type of errors as they contribute to faulty results. These errors are the most difficult to discover because they are not indicated by any messages. They usually result from an incorrect algorithm.

Visual Basic features embedded engines guaranteeing protection against and control of program execution errors.

Before an application is written, it is required to elaborate guidelines related to the purpose of a created program, and elaborate and check the algorithm for realization of the purpose. In order to fix an application for computations, it is required to design (as well as reserve sufficient memory resources for data) proper structures where our data are to be stored. These are mainly arrays and records. They are the structural types consisting of many elements. The arrays always contain elements of the same type (including a structural type) and records that can consist of elements of different types. Such action is called declaring. As for the arrays, it is possible to use dynamic declarations, i.e. change the array size up to the requirements.

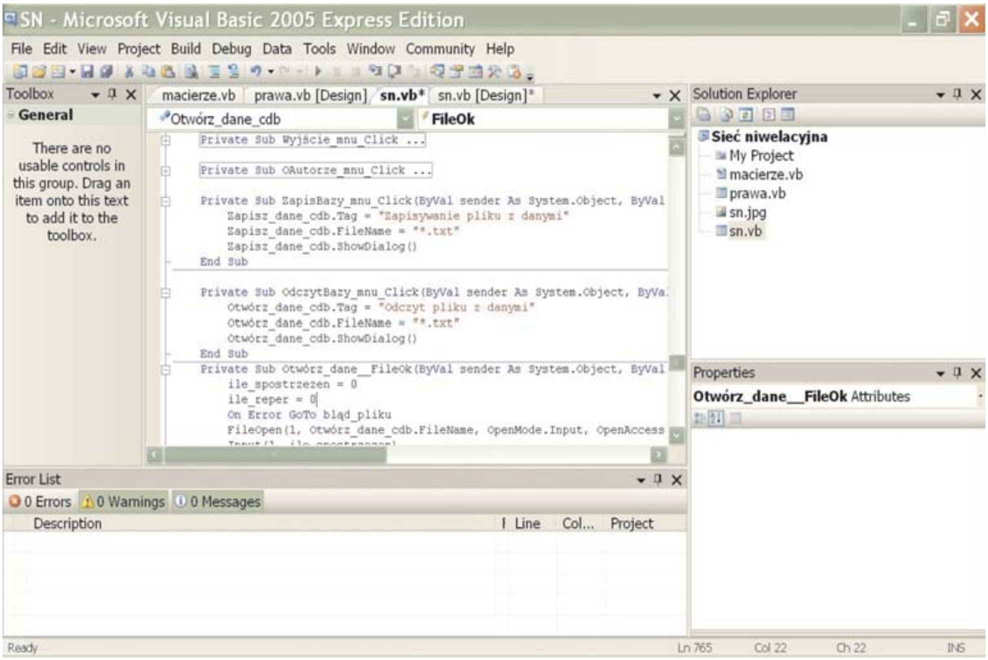


Properties

Lista_reperów System.Windows.Forms.ListB

(DataBindings)	
(Name)	**Lista_reperów**
AccessibleDescr	
AccessibleName	
AccessibleRole	Default
AllowDrop	False
Anchor	Top, Left
BackColor	□ Window
BorderStyle	Fixed3D
CausesValidatio	True

Fig. 1. The Visual Basic window featuring an element of design of the application for adjustment of a levelling net


SN - Microsoft Visual Basic 2005 Express Edition
File Edit View Project Build Debug Data Tools Window Community Help
Toolbox
General
There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.
macierze.vb prawa.vb [Design] sn.vb* sn.vb [Design]*
Otwórz_dane_cdb
FileOk
Solution Explorer
Sieć niwelacyjna
My Project
macierze.vb
prawa.vb
sn.jpg
sn.vb
Properties
Otwórz_dane__FileOk Attributes
Error List
0 Errors
0 Warnings
0 Messages
Description
Line
Col...
Project
Ready
Ln 765
Col 22
Ch 22
INS

```
    Private Sub Wyjście_mnu_Click ...

    Private Sub OAutorze_mnu_Click ...

    Private Sub ZapisBazy_mnu_Click(ByVal sender As System.Object, ByVal
        Zapisz_dane_cdb.Tag = "Zapisywanie pliku z danymi"
        Zapisz_dane_cdb.FileName = "*.txt"
        Zapisz_dane_cdb.ShowDialog()
    End Sub

    Private Sub OdczytBazy_mnu_Click(ByVal sender As System.Object, ByVa
        Otwórz_dane_cdb.Tag = "Odczyt pliku z danymi"
        Otwórz_dane_cdb.FileName = "*.txt"
        Otwórz_dane_cdb.ShowDialog()
    End Sub
    Private Sub Otwórz_dane__FileOk(ByVal sender As System.Object, ByVal
        ile_spostrzezen = 0
        ile_reper = 0
        On Error GoTo błąd_pliku
        FileOpen(1, Otwórz_dane_cdb.FileName, OpenMode.Input, OpenAccess
```

Fig. 2. The Visual Basic window featuring a part of the application code in the program code edition mode

An array, for which only the number of pointers is determined, and the element type are declared:

- *Dim matrixA (,) As Single* – i.e. two-pointer array containing the real type elements of a single precision; if an array is used, it is redeclared and the size is determined;
- *ReDim matrixA (w, k)* – as a new matrix;
- *ReDim Preserve matrixA (w, k)* – as a matrix of changed size with an unaltered content.

Thanks to the records it is possible to declare own complex types, i.e. the types that match geodetic data, e.g. Benchmark represented by number, height (estimated, calculated and adjusted), accuracy of height assessment, benchmark type (for adjustment):

- *Structure Height_points,*
- *Dim number As String,*
- *Dim est, cal, adj As Single,*
- *Dim error as single,*
- *Dim type as Char,*
- *End Structure.*

Such declared records can be used to declare an array:

Dim benchmark () As Height_points – i.e. single-pointer array containing the elements of height_points type (records).

Having such declared variables and constant values in our application, it is possible to start writing a program code.

3. An Application for Adjustment of a Levelling Net (within the Basic Scope)

Every person who knows programming bases can write a computer program making it possible to execute any process of geodetic computations. In order to create such application a proper knowledge for solution of the problem is required, i.e. the knowledge of an algorithm for solution of the problem is required.

Computation of a levelling net according to the algorithm of adjustment by means of the least squares method is presented as follows.

The basic formulas used in the program are listed below.

Observation weights

$$p_i = \frac{1}{m_i^2}.$$

The system of correction equations is generated using the following relation

$$H_p + h_i + v_i = H_N.$$

The correction equations in the matrix form

$$V = A \cdot x + w.$$

The solution of the system of standard equations

$$\begin{gathered}(A^T \cdot p \cdot A) \cdot X + (A^T \cdot p \cdot w) = 0 \\ -x = (A^T \cdot p \cdot A)^{-1} \cdot (A^T \cdot p \cdot w).\end{gathered}$$

Computation of adjusted coordinates

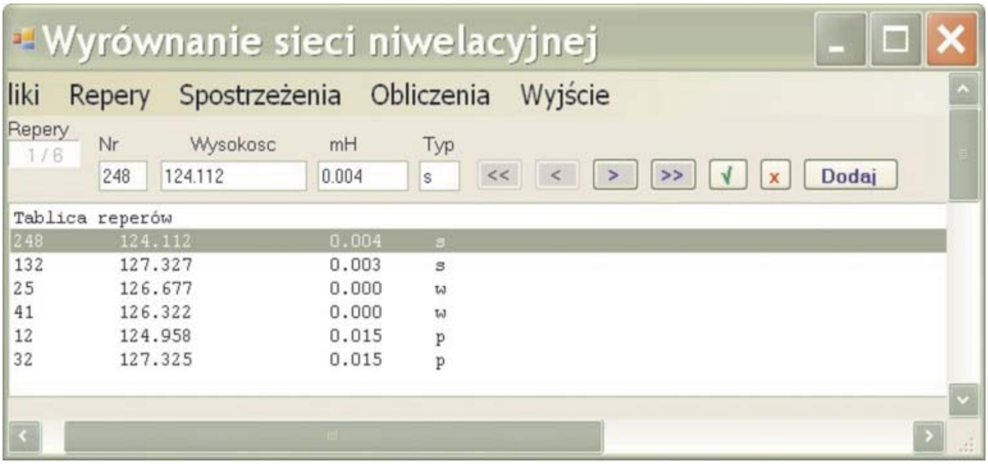
$$H_i = H_i^W + dH_i.$$

Basic analysis of accuracy

$$m_0 = \pm\sqrt{\frac{[pvv]}{n-u}}, \quad \operatorname{cov}(x) = m_0^2 \cdot (A^T \cdot p \cdot A)^{-1}.$$

Thanks to the written program code it is possible to display the results in a desired form and to save them in files (e.g. text).

The windows featuring the created application are shown in figures 3 and 4.



Repery 1 / 6

Nr 248

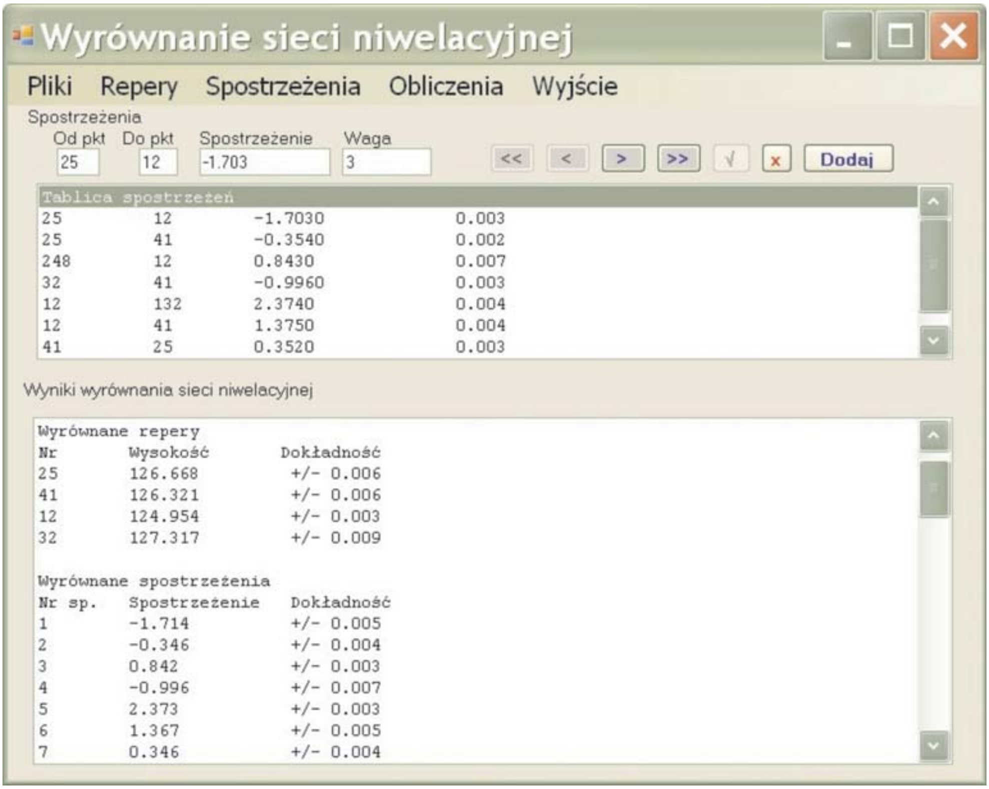
Wysokosc 124.112

mH 0.004

Typ s

```
Tablica reperów
248        124.112           0.004     s
132        127.327           0.003     s
25         126.677           0.000     w
41         126.322           0.000     w
12         124.958           0.015     p
32         127.325           0.015     p
```

Fig. 3. The start window of the finished application for adjustment of levelling nets



Spostrzeżenia

Od pkt: 25
Do pkt: 12
Spostrzeżenie: -1.703
Waga: 3

```
Tablica spostrzeżeń
25        12        -1.7030           0.003
25        41        -0.3540           0.002
248       12        0.8430            0.007
32        41        -0.9960           0.003
12        132       2.3740            0.004
12        41        1.3750            0.004
41        25        0.3520            0.003
```

Wyniki wyrównania sieci niwelacyjnej

```
Wyrównane repery
Nr        Wysokość        Dokładność
25        126.668          +/- 0.006
41        126.321          +/- 0.006
12        124.954          +/- 0.003
32        127.317          +/- 0.009

Wyrównane spostrzeżenia
Nr sp.    Spostrzeżenie    Dokładność
1         -1.714           +/- 0.005
2         -0.346           +/- 0.004
3         0.842            +/- 0.003
4         -0.996           +/- 0.007
5         2.373            +/- 0.003
6         1.367            +/- 0.005
7         0.346            +/- 0.004
```

Fig. 4. The result window of the application for adjustment of levelling nets

4. Summary

Creating applications by means of Visual Basic has a lot of advantages. A design can be developed (enhanced by extra functions) without any limitations.

The current versions of Visual Basic (Visual Basic Net 2005 or 2008) feature an extra advantage (as for computations), i.e. capacity of many numeric types can be doubled. E.g. 4 bytes instead of 2 are used by the Integer type, and 8 bytes instead of 4 are used by the Long type, etc.

Another advantage is the option of using Polish characters in the declared names of design elements (*Dim average as Single*, etc.).

The text files created during computations are very useful as they are very convenient to view and edit as well as they can be used in other programs.

A very important element is a proper decimal separator character (for a given Windows installation). Visual Basic requires "." (dot) as the decimal separator.

The programs created by means of the dot separator use the system separator during writing real values. If the system separator is “, ” (comma) used in the standard Windows installation, it can occur that a program writes decimal values with commas, and during the next reading the decimal parts are cut off (i.e. it is equivalent to rounding down an integer). It results from the rule of conversion of a text string into a number (a comma is an unacceptable character for numbers). A decimal separator (in Windows systems) can be changed by means of the control panel (regional settings or international settings – depending on the system version).

References

[1] Czaja J.: *Modele statystyczne w informacji o terenie*. Wydawnictwa AGH, Kraków 1996.

[2] MacDonald M.: *Visual Basic 2004. Wprowadzenie do programowania w NET*. Helion, Gliwice 2007.

[3] Hausbrandt S.: *Rachunek wyrównawczy i obliczenia geodezyjne. Tom 1 i 2*. PPWK, 1971.

[4] Hyde R.: *Profesjonalne programowanie. Część 1. Zrozumieć komputer*. Helion, Gliwice 2005.

[5] Hyde R.: *Profesjonalne programowanie. Część 2. Myśl niskopoziomowo, pisz wysokopoziomowo*. Helion, Gliwice 2006.

[6] Instrukcja techniczna O-1: *Ogólne zasady wykonywania prac geodezyjnych*. GUGiK, Warszawa 1988.

[7] Łoś A.: *Rachunek wyrównawczy*. Państwowe Wydawnictwo Naukowe, Warszawa 1973.

[8] Roman S., Petrusha R., Lomax P.: *VB.NET Almanach*. Helion, Gliwice 2003.