

РОЛЯТА НА ЗАДАЧИТЕ ПРИ ИЗУЧАВАНЕ НА СЪБИТИЙНО ПРОГРАМИРАНЕ С VISUAL BASIC В СРЕДНОТО УЧИЛИЩЕ

Коста А. Гъров, Стефка Й. Анева

Настоящата разработка е посветена на ролята на задачите при изучаване на събитийно програмиране с Visual Basic в средното училище. Предлага се в курса на обучение да се решават задачи от пет типа. Разглеждат се примерни задачи и решения на някои от тях.

Известно е, че учебната програма по Информатика за XI-XII клас [1] предвижда четири основни направления за обучение. Основно място се отделя на усвояването на една съвременна и широко използвана технология за визуално програмиране, т. нар. „програмиране, управлявано от събития“ в среда на графичен потребителски интерфейс. Основните принципи и възможности на този вид програмиране следва да бъдат конкретизирани и изложени чрез специфична среда и технология за автоматизиране на програмирането. За целта се препоръчва да бъде използван MS Visual Basic 6.0 по следните съображения:

- MS Visual Basic 6.0 е едновременно среда за създаване на програми, управлявани от събития, и език за програмиране;
- езикът за програмиране е неделима част от всички универсални програмни пакети за канцеларска дейност на фирмата Майкрософт (MS Word, MS Excel, MS Access и др.) като средство за повишаване на степента на автоматизация при тяхното използване;
- вероятността за конфликт между програмни продукти на различни фирми нараства, което може да доведе до нежелани проблеми при изучаване на принципите на новия вид програмиране;
- запознаването с нов език за програмиране дава възможност на учениците да осъзнаят многообразието на света и възпитава техния изследователски дух на свободен избор;
- МОН притежава определен брой лицензи за този програмен продукт, които се предоставят при поискване безплатно на училищата, желаещи да го използват.

Разбира се, когато някое училище разполага с подготвен учител и лицензиран софтуер, то може да се използва и друга среда за създаване на програми, управлявани от събития. Един подход, използващ езика Паскал, е показан в [2].

В [3] е подробно разписана учебна програма за модул “Събитийно програмиране с Visual Basic 6.0” с подробно разпределение на учебните часове. В тази програма от общо 87 учебни часа са предвидени 50 часа за упражнение. Ето защо считаме, че въпроса за ролята и мястото на задачите в модула е съществен за учителя, преподаващ предмета.

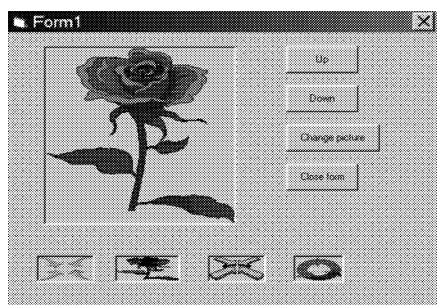
Ние предлагаме в курса на обучение да се решават 5 типа задачи:

- Първият тип задачи включва упражнение за усвояване на графичния потребителски интерфейс (ГПИ) и работа с основните елементи от него.
- Вторият тип задачи обучава учениците да работят със средствата и технологиите за анимация.
- Третият тип задачи е посветен на работата с масиви от елементи на ГПИ.
- Четвъртият тип задачи реализират потребителски настроени приложения за връзка с бази от данни (БД).
- Петият тип задачи включва създаване на стандартни Windows приложения, съдържащи менюта.

Поради ограничения обем на материала, тук ще разгледаме по-подробно задачите от първи и втори тип. За останалите типове задачи предлагаме общи сведения и примерни задачи без съответните решения. За читателите, които се интересуват от по-големи подробности по разглеждания проблем, ще препоръчаме списание “Математика и информатика”, в което през 2004 г. ще бъдат публикувани цикъл статии, посветени на задачите по модула “Събитийно програмиране”.

I. Задачи за усвояване на ГПИ и работа с основните елементи от него. Този тип задачи предвижда запознаване на учениците с онези елементи на ГПИ, които винаги присъстват в кутията с инструменти на Visual Basic (VB). За да изградим едно VB приложение (програма), трябва последователно да преминем през следните три етапа: 1. Рисуване на потребителския интерфейс; 2. Настройка на свойства на елементите на ГПИ; 3. Добавяне на програмен код към елементите. Всички решения на разглежданите по-нататък задачи следват тези три етапа. Пример за задача от първи тип е следната:

Задача. *Създайте приложение, което с помощта на два командни бутона увеличава и намалява размерите на дадена картинка (действието наподобява пускане и вдигане на щора). Прозорецът на приложението, както е показано на фиг. 1, съдържа още два командни бутона, с помощта на които се извършва съответно*



Фиг. 1

промяна на картинката и затваряне на формата и четири невидими елемента рисунка (PictureBox), които съхраняват заредените в тях графични изображения. Задайте име на проекта – shtora.vbp, име на форма – shtora.frm.

На учениците може да се предложи следното **решение**:

- 1) Създайте нов проект на Visual Basic;
- 2) Нарисувайте елементите на ГПИ върху формата (фиг. 1) и задайте описаните по долу в Таблица 1 свойства на елементите на ГПИ.

Таблица 1

Елементи на ГПИ	Име на елемент	Свойства
PictureBox	Picture1	Autosize=True; Picture – по избор
PictureBox	Picture2	Visible=False; Picture – по избор
PictureBox	Picture3	Visible=False; Picture – по избор
PictureBox	Picture4	Visible=False; Picture – по избор
PictureBox	Picture5	Visible=False; Picture – по избор
Commandbutton	CmdUp	Caption="Up"
Commandbutton	CmdDown	Caption="Down"
Commandbutton	CmdChange	Caption="Change picture"
Commandbutton	cmdClose	Caption="Close form"

Следва програмният код на VB:

```

Dim n As Integer
Dim s, p As Integer
Private Sub cmdChange_Click()
    If p = 0 Then
        Picture1.Picture = Picture2.Picture
        p = 1
    ElseIf p = 1 Then
        Picture1.Picture = Picture3.Picture
        p = 2
    ElseIf p = 2 Then
        Picture1.Picture = Picture4.Picture
        p = 3
    Else
        Picture1.Picture = Picture5.Picture
        p = 0
    End If
End Sub

Private Sub cmdClose_Click()
    End
End Sub
Private Sub cmdDown_Click()
    Do While s < n
        Picture1.Height = Picture1.Height + 1
        s = s + 1
    Loop
End Sub
Private Sub cmdUp_Click()
    n = Picture1.Height
    s = n
    Do While s > 50
        Picture1.Height = Picture1.Height - 1
        s = s - 1
    Loop
End Sub

```

II. Средства и технологии за анимация във Visual Basic. Проста анимация във VB може да бъде реализирана по три начина:

- Създаване на невидими елементи **Image**, в които при проектиране в свойство **Picture** (Рисунка) се подготвят изображенията, които при изпълнение ще се копират със същото свойство на видим елемент;
- Промяна на мястото (свойства **Top** и **Left**) на видим елемент;
- Показване и скриване (свойство **Visible**) на елементи.

Пример за задача от този тип е следната:

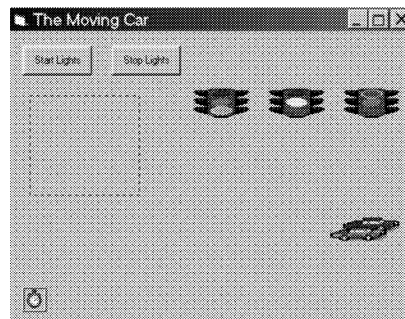
Задача. Създайте приложение, съдържащо следните елементи на ГПИ: два командни бутона (Commandbutton), един хронометър (Timer) и пет изображения (Image), както е показано на фиг. 2. Двата командни бутона във формата (фиг. 2) служат за активиране (бутон StartLights) и деактивиране на таймера (бутон StopLights). Три от картинките, на които са показани различните състояния на светофара (т.е. свойство **Picture** – съответно trffc10a.ico; trffc10b.ico; trffc10c.ico), са помощни и при стартирането на приложението не се виждат. За стойност на свойство Picture на четвъртия елемент Image е зададен файлът car.bmp. Петият елемент Image служи за визуализиране смяната на различните състояния

на светофара, която започва след натискането на бутона StartLights. При състояние на светофара светещо зелено, се стартира движението на колата с цел преминаване през светофара. Задайте име на проекта – CarTimer.vbp, име на форма – frmCarTimer.frm.

На учениците се предлага следното решение:

Създайте нов проект на Visual Basic;

Нарисувайте елементите на ГПИ върху формата (фиг. 2) и задайте описаните по долу в Таблица 2 свойства на елементите на ГПИ.



Фиг. 2

Таблица 2

Елементи на ГПИ	Име на елемент	Свойства
Form	frmCar	Caption="The Moving Car"
Commandbutton	cmdStartLights	Caption=" Start Lights"
Commandbutton	cmdStopLights	Caption="Stop Lights"
Image	imgLight	Stretch=True
Image	imgGreen	Stretch=True; Visible=false Picture=\\...\\trffc10a.ico
Image	imgYellow	Stretch=True; Visible=false Picture=\\...\\trffc10b.ico
Image	imgRed	Stretch=True; Visible=false Picture=\\...\\trffc10c.ico
Image	imgCar	Stretch=true; Picture=cars.ico
Timer	TmrLights	Enabled=false; Interval=1000

Забележка. Посочените графични файлове могат да бъдат намерени в Program Files\\Microsoft Visual Studio\\Common\\Graphics\\Icons.

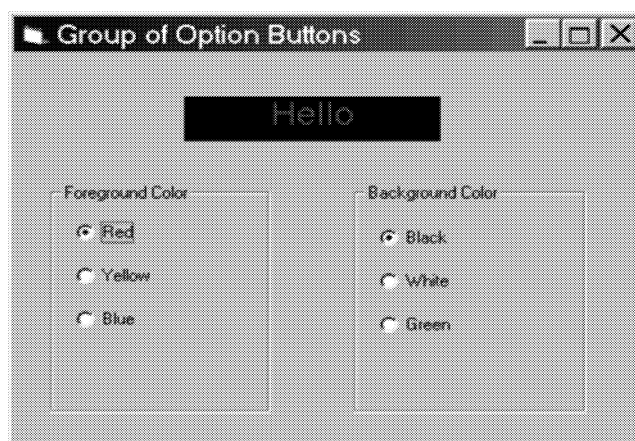
Следва програмният код на VB:

<pre>Option Explicit Dim original As Single Private Sub Form_Load() imgLight.Picture = imgYellow.Picture End Sub Private Sub tmrLights_Timer() Select Case imgLight.Picture Case imgGreen.Picture imgLight.Picture = imgYellow.Picture Case imgYellow.Picture imgLight.Picture = imgRed.Picture Case imgRed.Picture imgLight.Picture = imgGreen.Picture End Select End Sub</pre>	<pre>original = imgCar.Left Do While imgCar.Left > 0 imgCar.Left = imgCar.Left - 1 Loop imgCar.Left = original End Select End Sub Private Sub cmdStartLights_Click() tmrLights.Enabled = TRUE End Sub Private Sub cmdStopLights_Click() tmrLights.Enabled = FALSE End Sub</pre>
--	--

III. Работа с масиви от елементи на ГПИ. Често в дадено приложение се използват няколко елемента на ГПИ от един и същи клас от обекти и тези елементи реагират на едни и същи събития. В такива случаи вместо да се пише събитийна

процедура поотделно за всеки елемент, по-удобно е елементите да бъдат обединени в масив. Предимството при използване на масив от елементи на ГПИ е, че за всеки елемент от масива действа една и съща събитийна процедура, която е обща за всички елементи на масива. Отделните елементи на масива имат едно и също име, но се различават по своите индекси (т.е. стойностите на свойството им $\text{Index}(0, 1, 2, \dots)$). Пример за задача от този тип е следната:

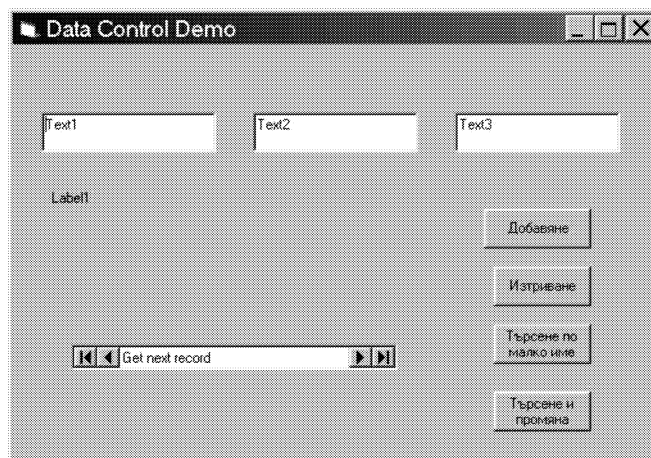
Задача. *Създайте приложение, което съдържа следните елементи на ГПИ: етикет (Label), две рамки (Frame), във всяка от които има група от три опционални бутона (Optionbutton), както е показано на фиг. 3. При избор на даден опционален бутон от рамката ForegroundColor се извършва настройка на свойството ForeColor (цвет на текста) на етикета. При избор на даден опционален бутон от рамката BackGroundColor се извършва настройка на свойството BackColor (фон) на етикета. Задайте име на проекта – Hello.vbp, име на форма – Hello.frm.*



Фиг. 3

IV. Задачи, реализиращи приложения за връзки с БД. С този тип задачи се изяснява въпросът за връзката на VB с БД. Ако желаете да създадете нов файл за база от данни, може да използвате както отделна програма за бази от данни (например Microsoft Access), така и допълнителната програма към Visual Basic, наречена Visual Data Manager. Пример за задачи от този тип е следната:

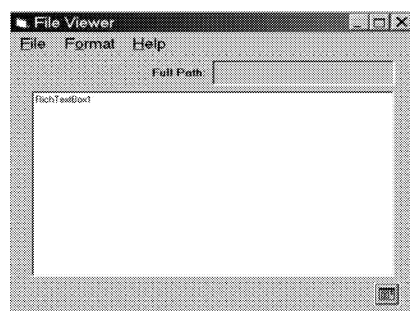
Задача. *Създайте приложение, съдържащо три текстови полета, четири командни бутона, един етикет и един елемент Data, както е показано на фиг. 4. Създайте базата данни Employees.mdb, съдържаща таблица EmployeeDetails, която се състои от три полета: FirstName, SurName и Salary. При преглеждане на различните записи в базата данни с помощта на елемента Data в съответните текстови полета да се показват данните за всеки запис от таблицата. Етикетът дава информация относно размера на заплатата за текущия запис. Четирите командни бутона осъществяват действията – добавяне, изтриване, търсене и промяна на записи в база данни. Задайте име на проекта – Datacontrol.vbp, име на формата – frmDatacontrol.frm.*



Фиг. 4

V. Задачи за приложения – тип меню. Менютата са важен елемент на формите. Те се проектират със специален редактор. За да създадете или редактирате менюта в дадено приложение на VB, трябва да работите с прозореца Menu Editor. Менютата разпознават само събитие **Click**. Това събитие възниква, когато потребителя щракне с бутона на мишката върху елемента от менюто. За да може изборът на елемент от потребителя да има някакви последствия, трябва да бъде поставен изпълним код в процедурата на събитието Click за този елемент. Менютата са подредени в йерархична структура. Елемент от меню сам по себе си може да представлява меню, когато се използва за визуализиране на елементи от подменю. Пример за задача от този тип е следната:

Задача. *Създайте приложение, което съдържа следните елементи на ГПИ: два етикета (Label), RichTextBox, CommonDialog (елемент, осигуряващ достъп до стандартните диалогови прозорци за отваряне и съхранение на файлове, смяна на шрифт и цвят), както е показано на фиг. 5. Към задачата реализирайте меню,*



Фиг. 5

включващо следните елементи:

- Меню **File** съдържа следните команди: **New**, **Open**, **Save** и **Exit**.
- Меню **Format** съдържа следните команди: **BackColor**, **ChangeFont**.
- Меню **Help** включва команда **About**.

Приложението има за цел да бъдат реализирани действията, свързани с преглеждане, форматиране и съхраняване на текстови файлове (формат *TXT* или *RTF*). Задайте име на проекта – *FileViewer.vbp*, име на формата – *FileViewer.frm*.

ЛИТЕРАТУРА

- [1] Учебни програми по информатика за X, XI, XII клас – второ равнище, *Математика и информатика*, **3–4**, 2001.
- [2] Б. Христов. Събитийно програмиране чрез Паскал, *Математика и информатика*, **4**, 2002.
- [3] Вл. С. ШКУРТОВ, К. А. ГЪРОВ, Ст. Й. АНЕВА. Изучаване на събитийно програмиране в средното училище. *Математика и математическо образование*, **32** (2003), 401–408.
- [4] УОЛЪС УАНГ. Visual Basic 6.0 за начинаещи. АлексСофт, С., 2000.
- [5] УОЛЪС УАНГ. Visual Basic 6. Ръководство за програмиста. АлексСофт, С., 2002.

Коста Андреев Гъров
бул. “Източен” 111
4000 Пловдив
e-mail: kosgar@pu.acad.bg

Стефка Йорданова Анева
ул. “Васил Чекеларов” 5
4004 Пловдив
e-mail: st_aneva@pu.acad.bg

THE ROLE OF THE PROBLEMS IN LEARNING OF EVENT-DRIVEN PROGRAMMING WITH VISUAL BASIC IN HIGH SCHOOLS

Kosta A. Garov, Stefka I. Aneva

The present paper deals with the role of the problems in learning of event-driven programming with Visual Basic in high schools. Problems of five types are offered in the time of training. A full solution of one problem is presented. Problems for individual work are offered.