

СИСТЕМА ОТ БАЗОВИ ЗАДАЧИ ЗА ИЗУЧАВАНЕ НА СЪБИТИЙНО ПРОГРАМИРАНЕ ЧРЕЗ СРЕДАТА VISUAL C# В СРЕДНОТО УЧИЛИЩЕ¹

гл. ас. Стефка Анева

ПУ „П. Хилендарски“, Пловдив, бул. България № 236
stfaneva@uni-plovdiv.bg

Настоящата разработка разглежда изучаването на събитийно програмиране чрез средата Visual C# в профилираната подготовка по информатика в средното училище. Посочена е примерна класификация на система от задачи за изучаване на учебното съдържание за модула „Събитийно програмиране в среда на графичен потребителски интерфейс“. Дискутирани са основните етапи, които трябва да се спазват при изпълнение на конкретна задача. Предложени са набор от базови задачи, целящи запознаване със средата на Visual C#, усвояване на графичния потребителски интерфейс и работа с основните елементи от него.

Ключови думи: събитийно програмиране, визуално програмиране, графичен потребителски интерфейс, елемент, събитие, събитийна процедура.

I. Въведение.

• Навлизането на компютрите и информационните технологии във всички сфери на живота налага съответни изисквания и към образователната система – тя трябва да реагира адекватно на необходимостта да се подготвят кадри, способни да използват и развиват новите технологии.

• В учебната програма по Информатика за XI-XII клас [1] 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 (ГПИ). **Основните принципи и възможности на този вид програмиране трябва да бъдат конкретизирани и изложени чрез специфична среда и технология за автоматизиране на програмирането. Запознаването с нов език за програмиране дава възможност на учениците да осъзнаят многообразието на света и възпитава техния изследователски дух на свободен избор.**

• Чрез настоящата разработка е предложен модел за организиране и провеждане на задължително-избираема подготовка по информатика за темата „Събитийно програмиране в среда на ГПИ“ чрез използване на средата Visual C#.

• В процеса на изучаване на темата „Събитийно програмиране в среда на графичен потребителски интерфейс“ трябва да бъдат разгледани следните пет основни теми:

- 1) Основни понятия на събитийното програмиране и ГПИ.
- 2) Основни елементи на езика за програмиране.
- 3) Обекти на ГПИ.
- 4) Взаимодействие с ОС.
- 5) Връзка с база от данни (БД).

II. Основни типове задачи за модула „Събитийно програмиране в среда на графичен потребителски интерфейс“

• В курса на обучение за модула „Събитийно програмиране в среда на графичен потребителски интерфейс“ трябва да бъдат разгледани следните 5 типа задачи:

1. Първият тип задачи включва набор от упражнения за усвояване на графичния потребителски интерфейс (ГПИ) и работа с основните елементи от него (**базови задачи**).
2. Вторият тип задачи е за формиране на знания и умения за реализиране на графика и анимация.
3. Третият тип задачи е посветен на работата с масиви от елементи на ГПИ и използване при необходимост от групиране на специални „елементи-носители“ (контейнери) на ГПИ.
4. Четвъртият тип задачи реализира потребителски настроени приложения за връзка с бази от данни.
5. Петият тип задачи включва създаване на стандартни Windows приложения, съдържащи менюта.

III. Базови задачи за усвояване на графичния потребителски интерфейс и работа с основните негови елементи.

В резултат от обучението на този базов етап ученикът трябва да:

- Знае да моделира и разработва графичен потребителски интерфейс с визуални средства;
- Умее да избира подходящ елемент на ГПИ в съответствие с необходимата функционалност на графичния интерфейс;
- Умее да настройва свойствата на елементите на ГПИ в режим на проектиране и режим на изпълнение;
- Знае да програмира подразбиращи се и други събития за основни елементи на ГПИ.
- Знае механизма за деклариране на променливи и умее да декларира и използва локални и глобални променливи.
- Знае механизма за извикване на една събитийна процедура в друга.

• В **базовите задачи** трябва да бъдат включени категории задачи, с които се акцентира върху запознаване с предназначението и възможностите на следните основни елементи (обекти) на ГПИ, които се използват често при изграждане на графичен потребителски интерфейс:

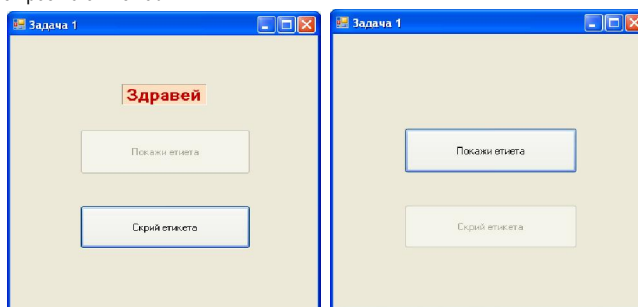
- **етикет** – за показване на текст (надписи) върху формата;
- **команден бутон** – за реализиране на действия (команди);
- **текстово поле** – за въвеждане на всякакви текстови данни в режим на изпълнение. Допълнително може да бъде осъществен контрол на въвеждането на данни в текстовото поле;

¹ Тази работа е подпомогната по проект ИС-М-4 на поделение „Научна и приложна дейност“ при Пловдивския университет „Паисий Хилендарски“.

- **изображение** – за визуализиране на графични изображения;
- **списъчна кутия** – за изобразяване на списък от символни низове за избор;
- **комбинирана кутия** – за редактиране на текст с възможност за избор и от падащ списък;
- **кутии с отметка** (контролни кутии) – за избор на независими по между си възможности от типа Да/Не;
- **рабиобутон** (изборен бутон) – за алтернативен избор на една от няколко взаимно изключващи се възможности.
- **Основни етапи при решаване на задачите:**
- За да се изгради едно приложение с ГПИ трябва последователно да бъдат изпълнени следните три етапа:
 - Рисуване на потребителския интерфейс на приложението;
 - Настройка на свойства на елементите на ГПИ;
 - Добавяне на програмен код към елементите (събитийни процедури).

Всички решения на разгледаните по-нататък задачи следват тези етапи.

Задача 1. Създайте приложение, съдържащо два командни бутона и един етикет, както е показано на фиг. 1. За свойство **Text** (надпис) на етикета задайте стойност "Здравей", а за двата бутона – съответно "Покажи етикета" и "Скрий етикета". При щракване върху бутон "Покажи етикета" се визуализира етикета с надпис "Здравей", а при щракване върху бутон "Скрий етикета" етикетът се скрива. В даден момент от изпълнение на задачата трябва да бъде достъпен само единият от тези два бутона, тъй като те изпълняват противоположни действия. При позициониране върху един от двата командни бутона, показалецът на мишката трябва да се променя в стрелка (т.е. във вида Arrow), а при позициониране върху формата – видът на показалеца на мишката трябва да бъде кръст (т.е. Cross). Съхранете проекта с име zad1.



Режим Изпълнение

Фиг. 1

- **Основни цели:**
- учениците да придобият начални знания и умения за работа с палитрата Toolbox с елементи на ГПИ;
- да разберат предназначението на свойствата Name и Text за разгледаните в задачата елементи на ГПИ – етикет (**Label**) и команден бутон (**Button**);
- да се запознаят с някои основни свойства, които са присъщи на повечето елементи на ГПИ – свойство видимост (Visible), свойство достъпност (Enabled), свойство фон на елемент (BackColor), свойство цвят на буквите на елемент (ForeColor) и др.;
- да придобият начални умения за създаване на Windows приложения в среда на Visual C#, спазвайки трите основни етапа;
- да придобият начални умения за промяна в режим на проектиране на стойностите на свойствата на елементите на ГПИ, зададени по подразбиране;
- да се запознаят с технологията за създаване на събитийни процедури за елементите на ГПИ.

На учениците се предлага следното решение:

- 1) **Първи етап:** Създаване на ГПИ на приложението (фиг. 1);
- 2) **Втори етап:** Настройка на някои свойства на елементите на ГПИ в режим на проектиране (таблица № 1).

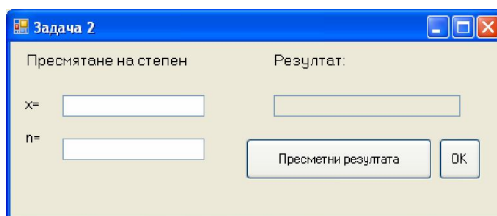
Таблица № 1

Елементи на ГПИ	Име на елемент	Свойства
Label	label1	Text="Здравей!"; Font, Bold, 14 BackColor – по избор
Button	buttonOn	Text="Покажи етикета"; Cursor=Arrow Enabled=false
Button	buttonOff	Text="Скрий етикета"; Cursor =Arrow Enabled=true
Form	Form1	Cursor =Cross; StartPosition=CenterScreen Text="Задача 1"

3) **Трети етап:** Добавяне на програмен код към елементите.

```
private void buttonOn_Click(object sender, EventArgs e)
{ label1.Visible = true; buttonOn.Enabled = false; buttonOff.Enabled = true; }
private void buttonOff_Click(object sender, EventArgs e)
{ label1.Visible = false; buttonOn.Enabled = true; buttonOff.Enabled = false; }
```

Задача 2. Създайте приложение за пресмятане на x^n . При въвеждане на стойности за x и n в текстови полета, с помощта на команден бутон **Пресметни резултата** да се извърши пресмятането и получения резултат да се визуализира в етикет (фиг. 2). С помощта на бутон **OK** да се изчиства въведената информация за x , n и получения вече резултат. Задайте име на създаденото от вас приложение - Zad2.



Фиг. 2

Основни цели:

- учениците да разберат предназначението и да се запознаят с някои свойства на разгледания в задачата нов елемент на ГПИ – **текстово поле (TextBox)**;
- да задълбочат знанията и уменията за работа с вече разгледаните в предишната задача елементи - етикет и команден бутон;
- да затвърдят формираните начални умения за създаване на Windows приложения в среда на Visual C#, спазвайки трите основни етапа;
- да затвърдят формираните умения за промяна в режим на проектиране на стойностите на свойствата на елементите на ГПИ, зададени по подразбиране;
- да се запознаят с начина за избор на активен (фокусиран) елемент на ГПИ в режим на изпълнение на приложението;
- да се запознаят с механизма за деклариране на променливи.

На учениците се предлага следното **решение**:

- 1) **Първи етап:** Създаване на ГПИ на приложението (фиг. 2);
- 2) **Втори етап:** Настройка на някои свойства на елементите на ГПИ в режим на проектиране (таблица № 2).

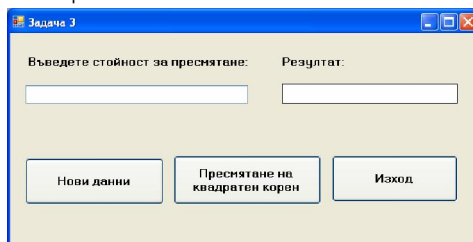
Таблица № 2

Елементи на ГПИ	Име на елемент	Свойства
Form	Form1	Text="Задача 2"
Label	label1	Text="Пресмятане на степен:"
Label	label2	Text="Резултат:"
Label	label3	Text="x="
Label	label4	Text="n="
TextBox	text_x	
TextBox	text_n	
TextBox	Result	BackColor=Control, ReadOnly="True"
Button	button1	Text="Пресметни резултата"
Button	button2	Text="ОК"

3) **Трети етап:** Добавяне на програмен код към елементите.

```
private void button1_Click(object sender, EventArgs e)
{
    double a,x,n;
    x = double.Parse(text_x.Text); n = double.Parse(text_n.Text);
    a = Math.Pow(x,n); result.Text = a.ToString();
}
private void button2_Click(object sender, EventArgs e)
{
    text_x.Text = ""; text_n.Text = ""; result.Text = ""; text_x.Focus();
}
```

Задача 3. Създайте приложение за пресмятане на квадратен корен на дадено число. При въвеждане на дадено число в текстово поле, с помощта на команден бутон да се пресмята квадратният корен на числото и полученият резултат да се визуализира в етикет (фиг. 3). Да се ограничи въвеждането на всички други символи от клавиатурата, различни от цифрите от 0 до 9 и десетичната запетая. Задайте име на създаденото от вас приложение - Zad3.



Фиг. 3

Основни цели:

- учениците да задълбочат своите познания и умения за работа с вече разгледаните в предишните задачи елементи – етикет, команден бутон и текстово поле;
- да развият своите умения за промяна в режим на проектиране на стойностите на свойствата на елементите на ГПИ, зададени по подразбиране;
- да умеят да декларират и използват локални променливи;
- да се запознаят с механизмите за преобразуване текст в число и обратно;
- да придобият начални знания и умения за ползване на логически оператор в C# и технологии за реализиране на диалог с потребителя по време на изпълнение на програмата;
- да се запознаят с технологията за създаване на събитийни процедури, различни от тези, които са зададени по подразбиране за даден елемент на ГПИ.

На учениците се предлага следното решение:

- 1) **Първи етап:** Създаване на ГПИ на приложението (фиг. 3);
- 2) **Втори етап:** Настройка на някои свойства на елементите на ГПИ в режим на проектиране (таблица № 3).

Таблица № 3

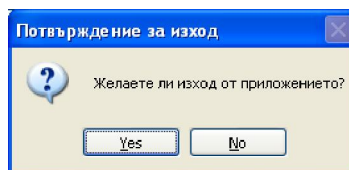
Елементи на ГПИ	Име на елемент	Свойства
Form	Form1	Text="Задача 3"
Label	label1	Text="Введете стойност за пресмятане:"
Label	label2	Text="Резултат:"
TextBox	txtInput	TextAlign=Right
Label	lblResult	Text=""; TextAlign= MiddleRight BorderStyle=FixedSingle ; BackColor=white
Button	cmdres	Text="Пресмятане на квадратен корен"
Button	cmdNew	Text="Нови данни"
Button	cmdExit	Text="Изход"

- 3) **Трети етап:** Добавяне на програмен код към елементите.

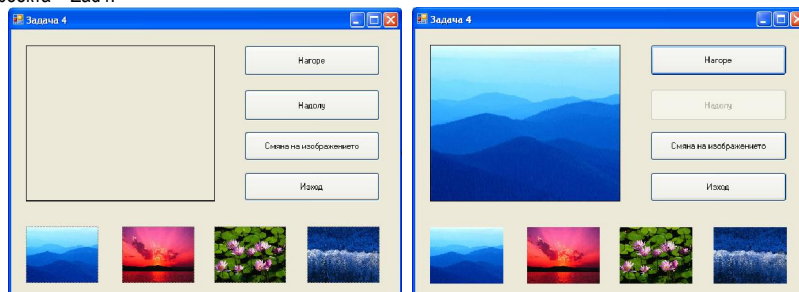
Пояснение: При въвеждане на данни от клавиатурата в режим на изпълнение често се налага да се извършва контрол на въведената информация. В разглежданата задача трябва да се ограничи в текстовото поле да се въвеждат от клавиатурата само неотрицателни реални числа, за да може да се пресметне квадратният корен.

```
private void cmdNew_Click(object sender, EventArgs e)
{ txtInput.Text = ""; lblResult.Text = ""; txtInput.Focus(); }
private void cmdResult_Click(object sender, EventArgs e)
{ double a;
  if (txtInput.Text!="")
  { a = double.Parse(txtInput.Text);
    a=Math.Sqrt(a); lblResult.Text = a.ToString(); }
  else
  MessageBox.Show ("Не сте въвели нищо в текстовото поле!");
}
private void cmdExit_Click(object sender, EventArgs e)
{ Close();}
private void txtInput_KeyPress(object sender, KeyPressEventArgs e)
{ if (((e.KeyChar < (char)Keys.D0) || (e.KeyChar > (char)Keys.D9))
  && !(e.KeyChar == (char)'.') && !(e.KeyChar == (char)Keys.Back))
  e.KeyChar = (char)0; }
```

Задача 4. Създайте приложение, което с помощта на два командни бутона увеличава и намалява размерите на дадена картина (PictureBox) (действието наподобява пускане и вдигане на щора). Прозорецът на приложението, както е показано на фиг. 4, съдържа още два командни бутона, с помощта на които се извършва съответно промяна на картината и затваряне на формата и още четири елемента PictureBox, които съхраняват заредените в тях графични изображения. При щракване върху бутон Изход да се визуализира следното съобщение:



Задайте име на проекта – Zad4.



Режим Проектиране

Режим Изпълнение

Фиг. 4

Основни цели:

- учениците да разберат предназначението и да се запознаят с някои свойства на разгледания в задачата нов елемент на ГПИ – рисунка (PictureBox);
- да задълбочат знанията си за работа с вече разгледания до момента елемент - команден бутон;
- да се запознаят с технологията за промяна на свойствата на елементите на ГПИ в режим на изпълнение;
- да се запознаят с механизма за деклариране на глобални променливи;
- да придобият начални знания и умения за ползване на оператори за цикъл в C#;
- да разберат предназначението на събитийната процедура Load за зареждане на форма.

На учениците се предлага следното решение:

- 1) **Първи етап:** Създаване на ГПИ на приложението (фиг. 4);
- 2) **Втори етап:** Настройка на някои свойства на елементите на ГПИ в режим на проектиране (таблица № 4).

Таблица №4

Елементи на ГПИ	Име на елемент	Свойства
Form	Form1	Text="Задача 4"
PictureBox	pictureBox1	BorderStyle=FixedSingle SizeMode=StretchImage
PictureBox	pictureBox2	Image – по избор; SizeMode=StretchImage
PictureBox	pictureBox3	Image – по избор; SizeMode=StretchImage
PictureBox	pictureBox4	Image – по избор; SizeMode=StretchImage
PictureBox	pictureBox5	Image – по избор; SizeMode=StretchImage
Button	cmdUp	Text="Нагоре"
Button	cmdDown	Text="Надолу"
Button	cmdChange	Text="Смяна на изображението"
Button	cmdExit	Text="Изход"

- 3) **Трети етап:** Добавяне на програмен код към елементите.

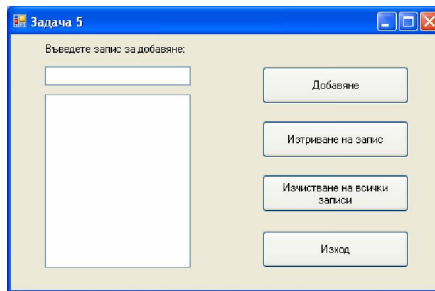
```

int p,n,s;
private void cmdUp_Click(object sender, EventArgs e)
{
    s = n;
    pictureBox1.SizeMode = PictureBoxSizeMode.Normal;
    while (s > 30)
    {
        pictureBox1.Height = pictureBox1.Height - 1; s = s - 1;
        cmdUp.Enabled = false; cmdDown.Enabled = true;
    }
private void cmdDown_Click(object sender, EventArgs e)
{
    pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;
    while (s < n)
    {
        pictureBox1.Height = pictureBox1.Height + 1; s = s + 1;
        cmdDown.Enabled=false; cmdUp.Enabled = true;
    }
private void cmdExit_Click(object sender, EventArgs e)
{
    if (MessageBox.Show("Желаете ли изход от приложението?",
        "Потвърждение за изход", MessageBoxButtons.YesNo,
        MessageBoxIcon.Question) == DialogResult.Yes) Application.Exit();
}
private void cmdChange_Click(object sender, EventArgs e)
{
    switch (p)
    {
        case 0:
            pictureBox2_Click(sender, e); break;
        case 1:
            pictureBox3_Click(sender, e); break;
        case 2:
            pictureBox4_Click(sender, e); break;
        case 3:
            pictureBox5_Click(sender, e); break;
    }
}
private void pictureBox2_Click(object sender, EventArgs e)
{
    pictureBox1.Image = pictureBox2.Image; p = 1;
}
private void pictureBox3_Click(object sender, EventArgs e)
{
    pictureBox1.Image = pictureBox3.Image; p = 2;
}
private void pictureBox4_Click(object sender, EventArgs e)
{
    pictureBox1.Image = pictureBox4.Image; p = 3;
}
private void pictureBox5_Click(object sender, EventArgs e)
{
    pictureBox1.Image = pictureBox5.Image; p = 0;
}
private void Form1_Load(object sender, EventArgs e)
{
    pictureBox1.Image = pictureBox2.Image;
    n = pictureBox1.Height; p = 1; cmdDown.Enabled = false;
}
    
```

Задача 5. Създайте приложение, което съдържа следните елементи на ГПИ: списъчна кутия, текстово поле, етикет и четири командни бутона, както е показано на фиг. 5. Действието на всеки един от бутоните да е следното:

- При въвеждане на произволна (недублираща се в списъчната кутия) стойност в текстовото поле и натискане на бутон *Добавяне* да се извършва добавяне на тази стойност в списъчната кутия.
- Бутон *Изтриване на запис* извършва изтриване на маркиран елемент от списъчната кутия.
- Бутон *Изчистване на всички записи* изтрива всички елементи от списъчната кутия.
- Бутон *Изход* затваря приложението.

Задайте име на проекта – Zad5.



Фиг. 5

Основни цели:

- учениците да разберат предназначението и да се запознаят с някои свойства на разгледания в задачата нов елемент на ГПИ – списъчна кутия (ListBox);
- да усвоят технологията за добавяне на елементи в списъчна кутия в режим на проектиране и в режим на изпълнение;
- да се запознаят с някои по-важни свойства и методи на елемента ListBox;
- да развият умения за ползване на логически оператор в C#.

На учениците се предлага следното решение:

- 1) **Първи етап:** Създаване на ГПИ на приложението (фиг. 5);
- 2) **Втори етап:** Настройка на някои свойства на елементите на ГПИ в режим на проектиране (таблица № 5).

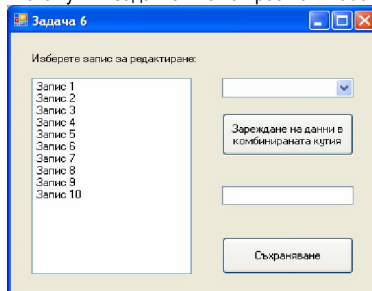
Таблица № 5

Елементи на ГПИ	Име на елемент	Свойства
Form	Form1	Text="Задача 5"
Label	Label1	Text="Въведете запис за добавяне:"
TextBox	txtInput	
ListBox	listBox1	
Button	cmdAdd	Text="Добавяне"
Button	cmdDelete	Text="Изтриване на избран запис"
Button	cmdClear	Text="Изчистване на всички записи"
Button	cmdExit	Text="Изход"

3) **Трети етап:** Добавяне на програмен код към елементите.

```
private void cmdExit_Click(object sender, EventArgs e)
{ Close(); }
private void cmdAdd_Click(object sender, EventArgs e)
{ if (txtInput.Text!="")
  { int index = listBox1.FindString(txtInput.Text, -1);
    if (index<0) {listBox1.Items.Add(txtInput.Text);}
    else
    {MessageBox.Show ("Елемента се повтаря!"); }
    txtInput.Text = ""; }
  txtInput.Focus(); }
private void cmdDelete_Click(object sender, EventArgs e)
{ listBox1.Items.Remove(listBox1.SelectedItem); }
private void cmdClear_Click(object sender, EventArgs e)
{ listBox1.Items.Clear(); }
private void txtInput_KeyPress(object sender, KeyPressEventArgs e)
{ if (e.KeyChar == (char)Keys.Return) cmdAdd_Click(sender, e); }
```

Задача 6. Създайте приложение, съдържащо следните елементи на ГПИ: етикет, списъчна кутия, комбинирана кутия, текстово поле и два командни бутона, както е показано на фиг. 6. При избор на даден елемент от списъчната кутия стойността му да бъде визуализирана в текстовото поле за редактиране. При натискане на бутона *Съхраняване* редактираната стойност в текстовото поле да се актуализира в списъчната кутия. При натискане на бутона *Зареждане на данни в комбинираната кутия* да се добавят в комбинираната кутия всички записи от списъчната кутия. Задайте име на проекта – Zad6.



Фиг. 6

Основни методически цели:

- учениците да разберат предназначението и да се запознаят с някои свойства и методи на разгледания в задачата нов елемент на ГПИ – комбинирана кутия (ComboBox);

- да усвоят технологията за добавяне на елементи в комбинирана кутия в режим на проектиране и в режим на изпълнение;
- да се запознаят с възможностите на събитийната процедура Load за зареждане на форма, с помощта на която в режим на изпълнение могат да бъдат автоматично добавени фиксиран брой елементи в списъчната кутия;
- да затвърдят своите знания относно някои важни свойства и методи на списъчните кутии, както и умения за работа с елемента ListBox;
- да развият умения за използване на оператор за цикъл.

На учениците се предлага следното решение:

- 1) **Първи етап:** Създаване на ГПИ на приложението (фиг. 6);
- 2) **Втори етап:** Настройка на някои свойства на елементите на ГПИ в режим на проектиране (таблица № 6).

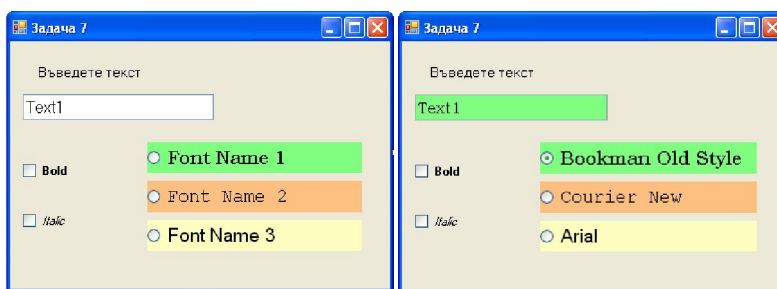
Таблица №6

Елементи на ГПИ	Име на елемент	Свойства
Form	Form1	Text="Задача 6"
ListBox	listBox1	
ComboBox	comboBox1	Items=Запис1, Запис2, Запис3, Запис4, Запис5 (всеки запис се въвежда на отделен ред)
Label	label1	Text="Изберете запис"
TextBox	txtEdit	Text=""
Button	cmdSave	Text="Съхраняване"
Button	cmdCombo	Text="Зареждане на данни в комбинираната кутия"

- 3) **Трети етап:** Добавяне на програмен код към елементите.

```
private void listBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    txtEdit.Text = listBox1.Text;
    listBox1.SelectedIndex = listBox1.FindString(txtEdit.Text);
}
private void cmdSave_Click(object sender, EventArgs e)
{
    listBox1.Items[listBox1.SelectedIndex]=txtEdit.Text;
}
private void Form1_Load(object sender, EventArgs e)
{
    listBox1.BeginUpdate();
    for (int x = 1; x <= 10; x++) { listBox1.Items.Add("Запис " + x.ToString()); }
    listBox1.EndUpdate();
}
private void cmdcombo_Click(object sender, EventArgs e)
{
    comboBox1.BeginUpdate();
    for (int x = 0; x <= listBox1.Items.Count-1; x++)
    {
        comboBox1.Items.Add(listBox1.Items[x]);
    }
    comboBox1.EndUpdate();
}
```

Задача 7. Създайте приложение, което съдържа етикет, текстово поле, три опционални бутона (RadioButton) и две полета за отметка (CheckBox), както е показано на фиг. 7. С помощта на опционалните бутони да се извършва избор на шрифт за текста, въведен в текстовото поле, както и настройка цвета на фона (свойство BackColor). Посредством полетата за отметка да се избира дали да бъде приложен или не стил Удебелен шрифт (Bold) и Курсивен шрифт (Italic) на въведения текст в текстовото поле. Задайте име на проекта – Zad7.



режим Проектиране

режим Изпълнение

Фиг. 7

Основни цели:

- учениците да разберат предназначението на разгледаните в задачата нови елементи на ГПИ – опционален бутон (RadioButton) и кутия за отметка (CheckBox);
- да се запознаят с някои по-важни свойства на елементите RadioButton и CheckBox;
- да могат да задават в процедурата Load за зареждане на форма начални инициализиращи настройки на свойства на елементи на ГПИ в режим на изпълнение.

На учениците се предлага следното решение:

- 1) **Първи етап:** Създаване на ГПИ на приложението (фиг. 7);
- 2) **Втори етап:** Настройка на някои свойства на елементите на ГПИ в режим на проектиране (таблица № 7).

Таблица №7

Елементи на ГПИ	Име на елемент	Свойства
Form	Form1	Text="Задача 7"
Label	label1	Text="Въведете текст"

TextBox	textBox1	Text="Text1"
CheckBox	checkBold	Text="Bold", Font – стил Bold
CheckBox	checkItalic	Text="Italic", Font – стил Italic
RadioButton	radioButton1	Text="Font Name 1"; BackColor – по избор Font – избор на шрифт Bookman Old Style
RadioButton	radioButton2	Text="Font Name 2"; BackColor – по избор Font – избор на шрифт Courier New
RadioButton	radioButton3	Text="Font Name 3"; BackColor – по избор Font – избор на шрифт Arial

3) Трети етап: Добавяне на програмен код към елементите.

```
private void radioButton1_CheckedChanged(object sender, EventArgs e)
{
    textBox1.Font = new Font(radioButton1.Font.Name, textBox1.Font.Size);
    textBox1.BackColor = radioButton1.BackColor;
}
private void checkItalic_CheckedChanged(object sender, EventArgs e)
{
    if (checkItalic.CheckState == CheckState.Checked)
    {
        if (checkBold.CheckState == CheckState.Checked)
            textBox1.Font = new Font(textBox1.Font, FontStyle.Italic | FontStyle.Bold);
        else
            textBox1.Font = new Font(textBox1.Font, FontStyle.Italic);
    }
    else
    {
        if (checkBold.CheckState == CheckState.Checked)
            textBox1.Font = new Font(textBox1.Font, FontStyle.Regular | FontStyle.Bold);
        else
            textBox1.Font = new Font(textBox1.Font, FontStyle.Regular);
    }
}
private void checkBold_CheckedChanged(object sender, EventArgs e)
{
    if (checkBold.CheckState == CheckState.Checked)
    {
        if (checkItalic.CheckState == CheckState.Checked)
            textBox1.Font = new Font(textBox1.Font, FontStyle.Bold | FontStyle.Italic);
        else
            textBox1.Font = new Font(textBox1.Font, FontStyle.Bold);
    }
    else
    {
        if (checkItalic.CheckState == CheckState.Checked)
            textBox1.Font = new Font(textBox1.Font, FontStyle.Regular | FontStyle.Italic);
        else
            textBox1.Font = new Font(textBox1.Font, FontStyle.Regular);
    }
}
private void radioButton2_CheckedChanged(object sender, EventArgs e)
{
    textBox1.Font = new Font(radioButton2.Font.Name, textBox1.Font.Size);
    textBox1.BackColor = radioButton2.BackColor;
}
private void radioButton3_CheckedChanged(object sender, EventArgs e)
{
    textBox1.Font = new Font(radioButton3.Font.Name, textBox1.Font.Size);
    textBox1.BackColor = radioButton3.BackColor;
}
private void Form1_Load(object sender, EventArgs e)
{
    radioButton1.Text = radioButton1.Font.Name;
    radioButton2.Text = radioButton2.Font.Name;
    radioButton3.Text = radioButton3.Font.Name;
    radioButton1.Checked = true;
}
```

IV. Заключение.

В настоящия материал е представен проект за организация на учебния процес и методика за провеждане на профилирано обучение по информатика за темата „Събитийно програмиране в среда на графичен потребителски интерфейс“, като за целта се използва средата на Visual C#. Предложени са 7 базови задачи за запознаване с основните възможности, които предоставя средата Visual C# за създаване на програми, управлявани от събития с ГПИ, включващ следните основни елементи – етикет, команден бутон, текстово поле, изображение, списъчна кутия, комбинирана кутия, контролна кутия и радиобутон.

През последните няколко години за преподаване на модула „Събитийно програмиране в среда на графичен потребителски интерфейс“ в средното училище се използваше предимно средата на Visual Basic 6.0, поради следните съображения:

- МОМН притежава определен брой лицензи за този програмен продукт, които се предоставят при поискване безплатно на училищата, желаещи да го използват;
- 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.

Интересни приложни проблеми и начини за тяхната оптимизация са дадени в [9].

Авторът на разработката има дългогодишен опит при обучението и квалификацията на бъдещи и настоящи учители по информатика и ИТ и множество методически разработки ([2], [3], [4], [5], [6], [7], [10]) за разглежданата тема с използване на средата Visual Basic.

Литература

- [1] МОИ, Дирекция „Политика в общото образование“, „Учебни програми III част за ЗП и ЗПП – IX, X, XI и XII клас“, София, 2003 г.
- [2] Шкуртов, Вл., К. Гъров, Ст. Анева, Изучаване на събитийно програмиране в средното училище, Сборник научни трудове, Тридесет и втора пролетна конференция на Съюза на математиците в България, Слънчев бряг 2003 г., стр. 401 - 408.
- [3] Шкуртов, Вл., К. Гъров, Ст. Анева, Учебна програма за изучаване на събитийно програмиране в средното училище, сп. „Математика и информатика“, кн. 6, 2003 г., стр. 8-12.
- [4] Гъров, К., Ст. Анева, Ролята на задачите при изучаване на събитийно програмиране с Visual Basic в средното училище, Сборник научни трудове, Тридесет и трета пролетна конференция на Съюза на математиците в България, Боровец 2004 г., стр. 322 - 328.

- [5] Гъров, К., Ст. Анева, Ролята и мястото на задачите в модула “Събитийно програмиране с Visual Basic 6.0”. Задачи за усвояване на основните елементи на графичния потребителски интерфейс, сп. “Математика и информатика”, приложение на кн. 1, 2004 г., стр. 1-31 (от приложението).
- [6] Гъров, К., Ст. Анева, За задачите в модула “Събитийно програмиране с Visual Basic 6.0”. Средства и технологии за анимация във Visual Basic. Работа с масиви от елементи на графичния потребителски интерфейс, сп. “Математика и информатика”, приложение на кн. 2, 2004 г., стр. 1-20 (от приложението).
- [7] Гъров, К., Ст. Анева, За задачите в модула “Събитийно програмиране с Visual Basic 6.0”. Задачи, реализиращи приложения за връзки с бази от данни. Задачи за приложения – тип меню, сп. “Математика и информатика”, приложение на кн. 3, 2004 г., стр. 1-20 (от приложението).
- [8] Либерти, Д., Visual C# 2005 Бележник на разработчика, ЗеСТ Прес, София, 2006 г.
- [9] Илиев, А., Н. Вълчанов, Т. Терзиева, Обобщения и оптимизации на някои алгоритми, Образованието в информационното общество, Пловдив, 12-13.5.2009 г., стр. 52-58.
- [10] http://old.fmi-plovdiv.org/vsh/VB6/VB6_Site.htm (посетен на 28.04.2010)