

Курс лабораторных работ по прикладному программированию

1. **Л.Р. 1** (2 часа): Основы программирования в среде **Delphi**: запуск Delphi; назначение компонентов главного окна: панель горячих кнопок, палитры компонентов, галерея шаблонов форм, окно инспектора объектов, окно формы, окно модуля. Проектирование оболочки программы-приложения: основное меню (**MainMenu**), всплывающее меню (**PopUpMenu**), их свойства, текстовые надписи на полотне.

* На полотне формы (Form1) разместить компонент **MainMenu** палитры компонентов **Standard** (двойной щелчок мыши на пиктограмме компонента). Открыть проектировщик меню (двойной щелчок мыши на компоненте **MainMenu1**). Заполнить пункты меню **Caption** (Заголовок) окне инспектора объектов (Object Inspector) :

&Ввод данных и дочерние меню:

&Ввод

&Контроль

&Диалог

&Расчет

&Выполнить

&Просмотр

&График

&Построить

в&Ыход

Листинг. 1.

(&Ы - означает инициализацию данного пункта меню дополнительно с помощью комбинации клавиш Alt+ы).

- Обработка события для меню **в&Ыход** (двойной щелчок на компоненте меню):

begin

close;

end;

Листинг. 2.

- Обработка события для остальных меню:

begin

ShowMessage('Простое сообщение');

end;

Листинг. 3.

- Поместить на полотне формы компонент **Static Text** палитры компонентов **Additional**. Изменить свойство **StaticText1** "Caption" на *Нажмите правую клавишу мыши*.
- Компонент **PopUpMenu1** :

&Нарисовать

&Очистить

&Изменить цвет фона

Изменить &цвет линии

Изменить &тип линии

Изменить тол&щину линии

Листинг. 4.

- ◇ Подключить всплывающее меню **PopUpMenu1** к полотну объекта Form1. Для этого щелкнуть левой клавишей мыши на полотне Form1 (в окне инспектора объектов заглавная надпись изменится на *Form1:TForm*); найти свойство *PopUpMenu* и изменить его значение на *PopUpMenu1*.
- ◇ Сохранить проект и файлы , протранслировать проект.

Л.Р. 2 (2 часа): графика на полотне Формы (объект Canvas, его свойства) и очистка полотна (кнопки "Построить" и "Очистить"), изменение цветовой палитры элементов графики (компонент **ColorDialog**).

- Обработка события для меню **&Нарисовать** :

```
begin
  with Canvas do
    begin
      moveTo(50,50);
      lineTo(ClientWidth-50,50);
      lineTo(ClientWidth-50,ClientHeight-50);
      lineTo(50,ClientHeight-50);
      lineTo(50,50);
    end;
  end;
```

Листинг . 5 .

или (результат будет почти такой же*):

```
begin
  Canvas.Rectangle( 50,50, Canvas.ClientWidth-50, Canvas.ClientHeight-50);
end;
```

Листинг . 6 .

- Обработка события для меню **&Очистить**:

```
begin
  Refresh ;
end;
```

Листинг . 7 .

- Изменение цветовой палитры : Поместить на полотно формы Form1 компонент **ColorDialog** из палитры компонентов **Dialogs**.

- ◇ Обработка события для меню **&Изменить цвет фона**:

```
begin
  colorDialog1.color:= form1.color;
  if colorDialog1.Execute then
    form1.color:= colorDialog1.color;
  end;
```

Листинг . 8 .

* Метод, приведенный в листинге 6, в отличие от предыдущего, заполняет область внутри построенного прямоугольника в стиле *Canvas.brush.Style* цветом *Canvas.brush.Color*.

- ◇ Обработка события для меню **Изменить &Цвет линии**:

```

begin
  colorDialog1.color:= Canvas.Pen.color;
  if colorDialog1.Execute then
    begin
      Canvas.Pen.color:= colorDialog1.color;
    end;
end;

```

Листинг. 9.

Л.Р. 3 (2 часа): графика на полотне Формы, контейнер для компонент (компонент **Panel**), визуализация окон и панелей, кнопки (**Button, BitBtn**), комбинированное диалоговое окно (**ComboBox**), обработка событий для кнопок.

- Создание диалога для пункта меню "Изменить &Тип линии ":
 - ◇ Поместить на полотно формы Form1 компонент **Panel** из палитры компонентов **Standard**.
 - ◇ Изменить свойство **Panel1** "Visible" на *False*.
 - ◇ Обработка события для меню **Изменить &Тип линии**.

```

begin
  Panel1.Visible:=true;
end;

```

Листинг. 10.

- ◇ Поместить на Панель1 компонент **ComboBox** (комбинированное диалоговое окно) из палитры компонентов **Standard** и два компонента **BitBtn** (стандартная кнопка с растровым значком) из палитры компонентов **Additional**.
- ◇ В окне инспектора объектов для **ComboBox1** изменить свойство "Style" на *csDropDownList* (ввод текста только из списка).
- ◇ В окне инспектора объектов для **ComboBox1** найти свойство "Items" и нажать на кнопку . Набрать в окне редактора списка строк (String List Editor) следующие строки:

```

сплошная
штриховая
пунктирная
штрих- пунктирная
штрих- пунктирная2
psClear
psInsideFrame

```

Листинг. 11.

(этот список будет отображаться в "падающей" части ListBox компонента ComboBox)

- ◇ Изменить свойство "Kind" для компонента **BitBtn1** на *bkOK*
- ◇ Изменить свойство "Kind" для компонента **BitBtn2** на *bkCancel*

- Обработка события для кнопки **ОК** (сохранить изменения)

```

begin
  Case ComboBox1.ItemIndex of
    0:      Canvas.Pen.Style:= psSolid ;
    1:      Canvas.Pen.Style:= psDash ;

```

```

2:      Canvas.Pen.Style:= psDot ;
3:      Canvas.Pen.Style:= psDashDot ;
4:      Canvas.Pen.Style:= psDashDotDot ;
5:      Canvas.Pen.Style:= psClear ;
6:      Canvas.Pen.Style:= psInsideFrame ;
end;
Panel1.Visible:=false;
end;

```

ЛИСТИНГ. 12.

- Обработка события для кнопки **Cancel** (отменить изменения)

```

begin
  Canvas.Pen.Style:= psSolid ;
  Panel1.Visible:=false;
end;

```

ЛИСТИНГ. 13.

Л.Р. 4 (2 часа): создание диалога, поля редактирования (**Edit**), создание собственных процедур, подключение других модулей (оператор **uses**), создание дочерних меню для всплывающего меню.

- Создание диалога для пункта меню " Изменить тол&Щину линии ":
 - ◇ Поместить на полотно формы Form1 компонент **Panel** из палитры компонентов **Standard**.
 - ◇ Изменить свойство **Panel2** "Visible" на *False*.
 - ◇ Поместить на Панель2 компонент **Edit** (поле редактирования) из палитры компонентов **Standard** и два компонента **BitBtn** из палитры компонентов **Additional**.
 - ◇ Изменить свойство "Kind" для компонента **BitBtn1** на *bkOK* .
 - ◇ Изменить свойство "Kind" для компонента **BitBtn2** на *bkCancel*.

- Обработка события для меню **Изменить тол&Щину линии**:

```

begin
  Panel2.Visible:=true;
end;

```

ЛИСТИНГ. 14.

- Обработка события для кнопки **ОК** (сохранить изменения):

```

begin
  Canvas.Pen.Width:=StrToInt(Edit1.Text);
  Panel2.Visible:=false;
  draw_kt_97 ;
end;

```

ЛИСТИНГ. 15.

- Обработка события для кнопки **Cancel** (отменить изменения):

```

begin
  Canvas.Pen.Width:=1;
  Panel2.Visible:=false;
  refresh;
  draw_kt_97 ;
end;

```

ЛИСТИНГ. 16.

- Оформить блок рисования (см. Листинг. 5, 6) в виде процедуры:

```
procedure TForm1.draw_kt_97 ;
begin
  Canvas.Rectangle( 50,50, Canvas.ClientWidth-50, Canvas.ClientHeight-50);
end;
```

Листинг. 17.

- ◇ Поместить ее в модуль **unit1** после операторов

```
implementation
{$R *.DFM}
var
```

Листинг. 18.

- ◇ В блоке описания процедур (TYPE) добавить:

```
procedure draw_kt_97 ;
```

Листинг. 19.

- ◇ Добавить вызов процедуры в обработчики событий изменения цвета и типа
линии (Листинг. 9, 12):

```
draw_kt_97 ;
```

Листинг. 20.

- Добавить в компонент **PopupMenu1** :

&Заполнение...

Листинг. 21.

и создать дочерние меню:

*ти&n
цв&ет*

Листинг. 22.

Для этого нажать правую клавишу мыши и выбрать из предложенного меню пункт:
Create Submenu (создать подменю).

- Обработка события для подменю **цв&Ет**:


```
begin
  colorDialog1.color:= form1.Canvas.Brush.color;
  if colorDialog1.Execute then
    begin
      form1.Canvas.Brush.color:= colorDialog1.color;
      draw_kt_97 ; end;
end;
```

Листинг. 23.

Л.Р. 5 (2 часа): графика на полотне Формы, заполнение замкнутых областей (класс Brush, его свойства), контейнер компонентов **GroupBox**, компонент **UpDown** (кнопка со стрелками для изменения числовой величины).

- Создание диалога для пункта подменю " ти&П ":

- ◇ Поместить на полотно формы Form1 компонент **GroupBox** из палитры компонентов **Standard**
- ◇ Изменить свойство **GroupBox1** "Visible" на *False*
- ◇ Изменить свойство **GroupBox1** "Caption" на *Выберите тип заполнения*
- ◇ Поместить на GroupBox1 компонент **ComboBox** (комбинированное диалоговое окно) из палитры компонентов **Standard** и две компоненты **BitBtn** (из палитры компонентов **Additional**

- ◇ В окне инспектора объектов для **ComboBox2** изменить свойство "Style" на *csDropDownList* (ввод текста только из списка)
- ◇ В окне инспектора объектов для **ComboBox2** найти свойство "Items" и нажать на кнопку  . Набрать в окне редактора списка строк (String List Editor) следующие строки:

Сплошная заливка
Очистка
Штриховка +45
Штриховка -45
Штриховка 0+90
Штриховка +45-45
Штриховка 0 (горизонтальная)
Штриховка 90 (вертикальная)

Листинг. 24.

- ◇ Изменить свойство "Kind" для компоненты **BitBtn1** на *bkOK*
- ◇ Изменить свойство "Kind" для компоненты **BitBtn2** на *bkCancel*

- Обработка события для подменю **ти&п**

```
begin
GroupBox1.Visible:=true;
end;
```

Листинг. 25.

- Обработка события для кнопки **ОК** (сохранить изменения) компоненты **GroupBox1**

```
begin
Case ComboBox2.ItemIndex of
0: Canvas.Brush.Style:= bsSolid ;
1: Canvas.Brush.Style:= bsClear ;
2: Canvas.Brush.Style:= bsBDiagonal ;
3: Canvas.Brush.Style:= bsFDiagonal ;
4: Canvas.Brush.Style:= bsCross ;
5: Canvas.Brush.Style:= bsDiagCross ;
6: Canvas.Brush.Style:= bsHorizontal ;
7: Canvas.Brush.Style:= bsVertical ;
end;
GroupBox1.Visible:=false;
draw_kt_97 ;end;
```

Листинг. 26.

- Обработка события для кнопки **Cancel** (отменить изменения) компоненты **GroupBox1**

```
begin
GroupBox1.Visible:=false;
end;
```

Листинг. 27.

- Использование компонента **UpDown** (кнопка со стрелками для изменения числовой величины):
- На полотне панели **Panel2** разместить компонент **UpDown** из палитры компонентов **Win32**
- ◇ Изменить свойство "Associate" для компонента **UpDown1** на *Edit1*

- ◇ Изменить свойство "Min" для компоненты **UpDown1** на *1*.
- ◇ Изменить свойство "Max" для компоненты **UpDown1** на *100*.
- ◇ Изменить свойство "Increment" для компоненты **UpDown1** на *1*.

Л.Р. 6 (2 часа): новая Форма, окно диалога, метки (Label), поля редактирования (Edit)

- Открыть новую форму (Form2), используя меню **File/New**. Выбрать из многостраничной палитры шаблонов форм галерею **Dialog**, а из этой галереи шаблон **Standard Dialog**. Полученное окно диалога получит имя *BtnBottomDlg* (или *OKBottomDlg1* в версии Delphi 4).

- Разместить на полотне *Bevell* вертикально пять меток **Label** из палитры компонентов **Standard** и подписать их соответственно, изменяя свойство *Caption* :

Размер области по X, pixel :

Размер области по Y, pixel :

Число элементов по X :

Число элементов по Y :

Базовый размер элемента :

Листинг. 28.

- Разместить на полотне *Bevell* вертикально пять полей редактирования *Edit* из палитры компонентов **Standard** , им будут автоматически присвоены имена *Edit1,...,Edit5*. Задать начальные значения для этих полей в свойстве *Text* соответственно 400, 400, 5, 5, 20.
- Рядом с полями *Edit* разместить пять компонент **UpDown** из палитры компонентов **Win32** и “привязать” их к полям *Edit1,...,Edit5*.
- Обработка события для кнопки **OK** :

var

xa, ya, nx_el, ny_el, nnn: integer;

begin

xa:=StrToInt(x_area.Text);

ya:=StrToInt(y_area.Text);

nx_el:=StrToInt(nx.Text);

ny_el:=StrToInt(ny.Text);

nnn:=StrToInt(size_el.Text);

BtnBottomDlg.Close;

form1.draw_area(xa, ya, nx_el, ny_el, nnn);

end;

Листинг. 29.

- Обработка события для кнопки **Cancel** :

begin

BtnBottomDlg.Close;

end;

Листинг. 30.

- «Привязать» открытие окна диалога к одному из пунктов основного меню, например:

Ввод \ Диалог

- Обработка события для меню *Диалог*:

begin

BtnBottomDlg.Show{диалог для рисования периодической структуры};

end;

Листинг. 31.

- В модуле *Unit1* в блоке **uses** описать подключение модуля *Unut2*.
- В модуле *Unit2* (соответствующем окну диалога *BtnBottomDlg*) описать подключение модуля *Unut1* после операторов*

implementation

*{ \$R *.DFM }*

uses unit1;

Листинг. 32.

* подключение модуля *Unut2* в блоке **uses** модуля *Unit1* позволяет обращаться из модуля *Unit1* к процедурам модуля *Unut2*, но не наоборот. Подключение модуля *Unit1* в блоке **uses** модуля *Unut2* в данном случае недопустимо.

- В модуле *Unit1* описать заголовок процедуры рисования в блоке **type** перед операторами описания других процедур:

```
procedure draw_area(xa, ya, nx_el, ny_el, nnn : integer);
```

ЛИСТИНГ. 33.

- В модуле *Unit1* в блоке **implementation** написать текст процедуры рисования:

```
procedure TForm1.draw_area(xa, ya, nx_el, ny_el, nnn : integer);
```

```
var
```

```
i,j,x1,y1,x2,y2,x_step,y_step :integer;
```

```
begin
```

```
form1.refresh;
```

```
form1.ClientWidth:=xa;
```

```
form1.ClientHeight:=ya;
```

```
x_step:= xa div nx_el; y_step:= ya div ny_el;
```

```
for i:=1 to nx_el do begin
```

```
x1:=(i-1)*x_step + x_step div 2 - nnn ; x2:=x1+nnn ;
```

```
for j:=1 to ny_el do begin
```

```
y1:=(j-1)*y_step + y_step div 2 - nnn ; y2:=y1+nnn ;
```

```
form1.Canvas.ellipse(x1,y1,x2,y2); {Для рисования окружностей и эллипсов}
```

```
{ form1.Canvas.polygon ( [ Point(x1,y1), Point(x2,y2), Point(x3,y3),..., Point(xn,yn) ] ); }
```

```
{ Для рисования замкнутых многоугольников }
```

```
end;
```

```
end;
```

```
end;
```

ЛИСТИНГ. 34.

Л.Р. 7 (1 час): контроль данных, изменение цвета визуальных компонент.

- Добавить в окне диалога *BtnBottomDlg* кнопку **Button** из палитры компонентов **Standard** и подписать ее: “Контроль”.
- Добавить в окне диалога *BtnBottomDlg* метку **Label** из палитры компонентов **Standard** и подписать ее: “Контроль не производился”.
- Обработка события для кнопки **Контроль**:

```
var
```

```
x_k,y_k:integer;
```

```
begin
```

```
xa:=StrToInt(x_area.Text);
```

```
ya:=StrToInt(y_area.Text);
```

```
nx_el:=StrToInt(nx.Text);
```

```
ny_el:=StrToInt(ny.Text);
```

```
nnn:=StrToInt(size_el.Text);
```

```
x_k:=nnn*2* nx_el;
```

```
y_k:=nnn*2* ny_el;
```

```
if (x_k > xa) or (y_k > ya) then begin
```

```
label7.caption:='Данные некорректны';
```

```
label7.color:=clRed ; end
```

```
else begin
```

```
label7.caption:='Данные корректны';
```

```
label7.color:=clGreen;
    end;
end;
```

Листинг. 35.

Л.Р. 8 (2 часа): Описание, открытие и закрытие текстового файла, чтение и запись данных. Окно диалога `OpenDialog`.

- Добавить в окне диалога `BtnBottomDlg` кнопку **Button** из палитры компонентов **Standard** и подписать ее: “Считать из файла”.
- Добавить в окне диалога `BtnBottomDlg` метку **Label** из палитры компонентов **Standard** и подписать ее: “Контроль файла”.
- В блоке описания глобальных переменных в модуле `Unit1` или `Unit2` добавить

```
var
...
tmp_str: string;
inp_fl: text;
...
implementation
```

Листинг. 36.

- Поместить на полотно формы `Form1` компонент **OpenDialog** из палитры компонентов **Dialogs**.
- Изменить свойство этого компонента `Filter` на следующее:

```
Все файлы|*.*|Файлы данных|*.dat|Текстовые файлы|*.txt
```

Листинг. 37.

- Обработка события для кнопки **Считать из файла**:

```
begin
    if form1.OpenDialog1.execute then begin
        tmp_str := form1.OpenDialog1.FileName;
        BtnBottomDlg.Label6.Caption := tmp_str;
        AssignFile(inp_fl, form1.OpenDialog1.FileName);
        reset(inp_fl);
        Readln(inp_fl, xa);
        X_area.Text := IntToStr(xa);
        Readln(inp_fl, ya);
        y_area.Text := IntToStr(ya);
        Readln(inp_fl, nx_el);
        nx.Text := IntToStr(nx_el);
        Readln(inp_fl, ny_el);
        ny.Text := IntToStr(ny_el);
        Readln(inp_fl, nnn);
        size_el.Text := IntToStr(nnn);
        system.close(inp_fl);
    end;
end;
```

(2-й семестр)

Л.Р. 1 (2 часа): основное меню (MainMenu), всплывающее меню (PopUpMenu), метки (Label), поля редактирования (Edit).

Л.Р. 2 (2 часа): кнопки (Button), графика на полотне Формы (объект Canvas, его свойства) и очистка полотна (кнопки "Построить" и "Очистить")

Л.Р. 3 (2 часа): назначение цвета линиям рисования и фону (компонента ColorGrid, объект Pen.***, свойство Color), назначение типа линий (объект Pen.***, свойство Style), назначение толщины линии (объект Pen.***, свойство Width), диалог.

1. Основы программирования в среде Delphi: запуск Delphi; назначение компонент главного окна: панель горячих кнопок, палитры компонент, галерея шаблонов форм, окно инспектора объектов, окно формы, окно модуля.

Меню

типовые для оболочек языковых средств "всплывающие" меню:

File

Edit

Search

View

Project

Run

Tools

Help

нетипичные меню:

Component

DataBase

Панель горячих кнопок

Выполнена в виде набора пиктограмм. Стандартный набор содержит:

Функции меню **File:**

- Open project
- Open file
- Save all
- Save file
- Add file to project
- Remove file from project
- New Form

Функции меню **View:**

- Select unit from list - *Выбор модуля из списка*
- Toggle Form/Unit - *Переключение между окнами Формы и Модуля*
- Select form from list - *Выбор формы из списка*

Функции меню **Run:**

- Run
- Trace into
- Step over
- Pause

Нажатие правой клавиши мыши на пиктограмме горячей клавиши инициализирует окно настройки, включающее операции:

- Show Hints - *показать*
- Hide - *скрыть*
- Help - *подсказка*
- Properties - *свойства*

Палитра компонент

