

unit dblpic_;

interface

uses

*Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, jpeg, ExtCtrls, Menus;*

type

```
TForm1 = class(TForm)
  Timer1: TTimer;
  MainMenu1: TMainMenu;
  N1: TMenuItem;
  N2: TMenuItem;
  N3: TMenuItem;
  N4: TMenuItem;
  procedure FormCreate(Sender: TObject);
  procedure FormPaint(Sender: TObject);
  procedure FormMouseDown(Sender: TObject; Button:
TMouseButton; Shift: TShiftState; X, Y: Integer);
  procedure Timer1Timer(Sender: TObject);
  procedure N1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
```

7. // объявление нового типа col_row

```
col_row = record
  col: integer;
  row: integer;
end;
```

const

```
MAX_SIZE = 32; // максимальное кол-во парных картинок
MAX_H = 8; // максимальный размер поля - 8x8
MAX_W = 8;
```

var

```
Form1: TForm1;
```

8. Pole: array [1..MAX_H,1..MAX_W] of integer;

```
{ Pole[i,j] < 100 - код картинки, клетка закрыта;
  Pole[i,j] > 100 и < 200 - клетка открыта, т.е. игрок видит картинку;
  Pole[i,j] > 200 - игрок нашел пару для этой картинки }
```

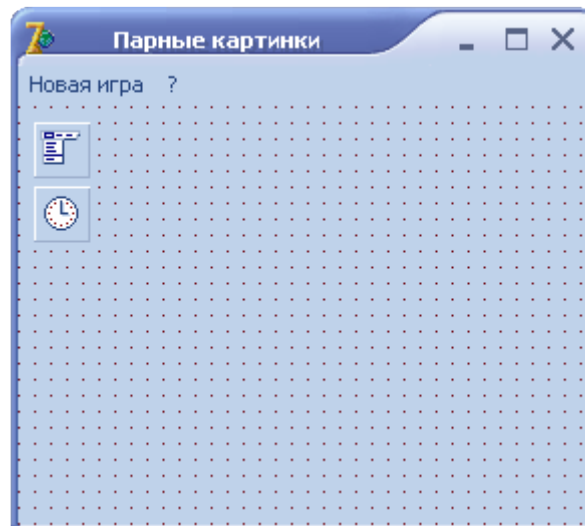
```
Pictures: TBitmap; // картинки, загруженные из файла
```

```
n : integer; // колво открытых пар картинок
```

```
count: integer; // количества открытых в данный момент клеток
```

```
open1: col_row; // координаты 1-й открытой клетки
```

```
open2: col_row; // координаты 2-й открытой клетки
```



W: integer; // кол-во клеток по горизонтали
H: integer; // кол-во клеток по вертикали. Произведение W и H должно быть кратно 2-м
WK: integer; // ширина клетки
HK: integer; // высота клетки

implementation

*{SR *.dfm}*

```
6. // рисует клетку поля
procedure Kletka(col,row: integer);
var
  x,y: integer; // левый верхний угол клетки (координаты)
  src, dst : Trect; // источник и получатель битового образа

begin
  // преобразуем координаты клетки в координаты на поверхности формы
  x := (col-1)*WK;
  y := (row-1)*HK;

  if Pole[col,row] > 200 then
    // для этой клетки найдена пара
    // клетку надо убрать с поля
    begin
      // установить цвет границы, закраски и текста
      Form1.Canvas.Brush.Color := clBtnFace;
      Form1.Canvas.Pen.Color := clBtnFace;
      Form1.Canvas.Font.Color := clBtnFace;
    end;

  if (Pole[col,row] > 100) and (Pole[col,row] < 200)
  then
    // клетка открыта - вывести картинку
    begin

      // Pole[col,row] = номер картинки + 100, где 100 - признак того, что клетка открыта
      // определим положение картинки в Pictures
      src := Bounds((Pole[col,row]-100 -1 )*WK,0,WK,HK);

      // координаты картинки (клетки) на форме
      dst := Bounds(x,y,HK-2,WK-2);

      // вывести картинку в клетку
      Form1.Canvas.CopyRect(dst,Pictures.Canvas,src);

      // установить цвет границы и цифры
      Form1.Canvas.Pen.Color := clBlack;
      Form1.Canvas.Font.Color := clBlack;
      Form1.Canvas.Brush.Style := bsClear;
    end;

  if (Pole[col,row] > 0) and (Pole[col,row] < 100) then
    // клетка закрыта, рисуем только контур
    begin
      Form1.Canvas.Brush.Color := clBtnFace;
      Form1.Canvas.Pen.Color := clBlack;
```

```

Form1.Canvas.Font.Color := clBtnFace;
end;

// отрисовать клетку
Form1.Canvas.Rectangle(x,y,x+WК-2,y+HK-2);
//Form1.Canvas.TextOut(x+15,y+15, IntToStr(Pole[col,row]));
Form1.Canvas.Brush.Color := clBtnFace;
end;

// отрисовывает поле
procedure ShowPole;
var
    row,col: integer;
begin
    for row:=1 to H do
        for col:=1 to W do
            Kletka(row,col);
        end;
    end;
end;

// новая игра
Procedure NewGame;
var
    k: integer;    // кол-во парных картинок
    r: integer;    // случайное число
    buf: array[1..MAX_SIZE] of integer;
    // в buf[i] записываем, сколько чисел i
    // записали в массив Pole
    i,j: integer; // индексы массивов
begin
    Randomize;
    k := Trunc(H*W/2);

    for i:=1 to k do
        buf[i] := 0;
    end;

    // запишем в массив Pole случайные числа от 1 до 2. Каждое число должно быть записано два раза
    for i:=1 to H do
        for j:=1 to W do
            begin
                repeat
                    r := random (k) + 1;
                until buf[r] < 2;
                Pole[i,j] := r; // код картинки
                inc(buf[r]);
            end;
        end;
    end;
    // здесь поле сгенерировано
    n:=0;
    ShowPole;
end;

// создание формы
1. procedure TForm1.FormCreate(Sender: TObject);
var
    np: integer; // кол-во парных картинок
begin
    Pictures := TBitmap.Create;

```

```

// загрузить картинки из файла
Pictures.LoadFromFile('pictures.bmp');

НК := Pictures.Height-1; // высота картинки
WK := НК;           // ширина картинки

np:= Round(Pictures.Width / WK);
if np <= 15
  then H := 4
  else H :=5;
W := Round(np*2/H);

// установить размера поля
Form1.ClientHeight := H * НК;
Form1.ClientWidth := W * WK;

Form1.Timer1.Enabled := False;
Form1.Timer1.Interval := 200;

n := 0;
NewGame;
end;

// прорисовка клеток на поле
2. procedure TForm1.FormPaint(Sender: TObject);
begin
  ShowPole;
end;

// щелчок в клетке
3. procedure TForm1.FormMouseDown(Sender: TObject; Button: TMouseButton; Shift: TShiftState; X, Y: Integer);
var
  col_: integer; // номер клетки по горизонтали
  row_: integer; //номер клетки по вертикали

begin
  col_ := Trunc(X/WK) + 1;
  row_ := Trunc(Y/НК) + 1;

  if Pole[col_,row_] > 200 then
    // щелчок в на месте одной из двух уже найденных парных картинок
    exit;

  // открытых клеток нет
  if count = 0 then
    begin
      count := 1;
      open1.col := col_;
      open1.row := row_;

      // клетка помечается как открытая
      Pole[open1.col,open1.row] := Pole[open1.col,open1.row] + 100;
      Kletka(open1.col,open1.row);
      exit;
    end;

```

```

// открыта одна клетка, надо открыть вторую
if count = 1 then begin
  open2.col := col_;
  open2.row := row_;

  // если открыта одна клетка и щелчок сделан в этой клетке, то ничего не происходит
  if (open1.col = open2.col) and (open1.row = open2.row)
    then exit

  else begin
    count := 2; // теперь открыты две клетки
    Pole[open2.col,open2.row] :=
      Pole[open2.col,open2.row] + 100;
    Kletka(open2.col,open2.row); // рисуем вторую клетку

    // проверим, открытые картинки одинаковые?
    if Pole[open1.col,open1.row] = Pole[open2.col,open2.row] then
      // открыты две одинаковые картинки
      begin
        n := n+1;
        Form1.Timer1.Enabled := True; // запустить таймер
        // процедур обработки события OnTimer "сотрет" две одинаковые картинки
      end;
    end;
  exit;
end;

if count = 2 then
begin
  // открыты 2 клетки с разными картинками. Закроем их и откроем новую, в которой сделан щелчок
  // закрыть открытые клетки
  Pole[open1.col,open1.row] := Pole[open1.col,open1.row] - 100;
  Pole[open2.col,open2.row] := Pole[open2.col,open2.row] - 100;
  Kletka(open1.col,open1.row);
  Kletka(open2.col,open2.row);

  // запись в open1 номера текущей клетки
  open1.col := col_;
  open1.row := row_;
  count := 1; // счетчик открытых клеток

  // открыть текущую клетки
  Pole[open1.col,open1.row] := Pole[open1.col,open1.row] + 100;
  Kletka(open1.col,open1.row);
end;
end;

// обработка события таймера
4. procedure TForm1.Timer1Timer(Sender: TObject);
begin
  // в массиве Pole клетки помечаются как совпавшие
  Pole[open1.col,open1.row] := Pole[open1.col,open1.row] + 100;
  Pole[open2.col,open2.row] := Pole[open2.col,open2.row] + 100;
  count := 0;

```

```
// отрисовать клетки
Kletka(open2.col,open2.row);
Kletka(open1.col,open1.row);

// остановка таймера
Form1.Timer1.Enabled := False;

if n = Trunc(W*H/2)
then // открыты все пары
begin
  Form1.Canvas.Font.Name := 'Times New Roman';
  Form1.Canvas.Font.Size := 36;
  Form1.Canvas.Font.Color := clBlack;
  Form1.Canvas.TextOut(70,160,'Game Over!');
  Form1.Canvas.Font.Size := 10;
  Form1.Canvas.TextOut(120,210,'(с) Культин Н.Б., 2003');
end;
end;

// выбор в меню команды Новая игра
5. procedure TForm1.N1Click(Sender: TObject);
begin
  Canvas.Rectangle(0,0,ClientWidth,ClientHeight);
  NewGame;
end;

end.
```