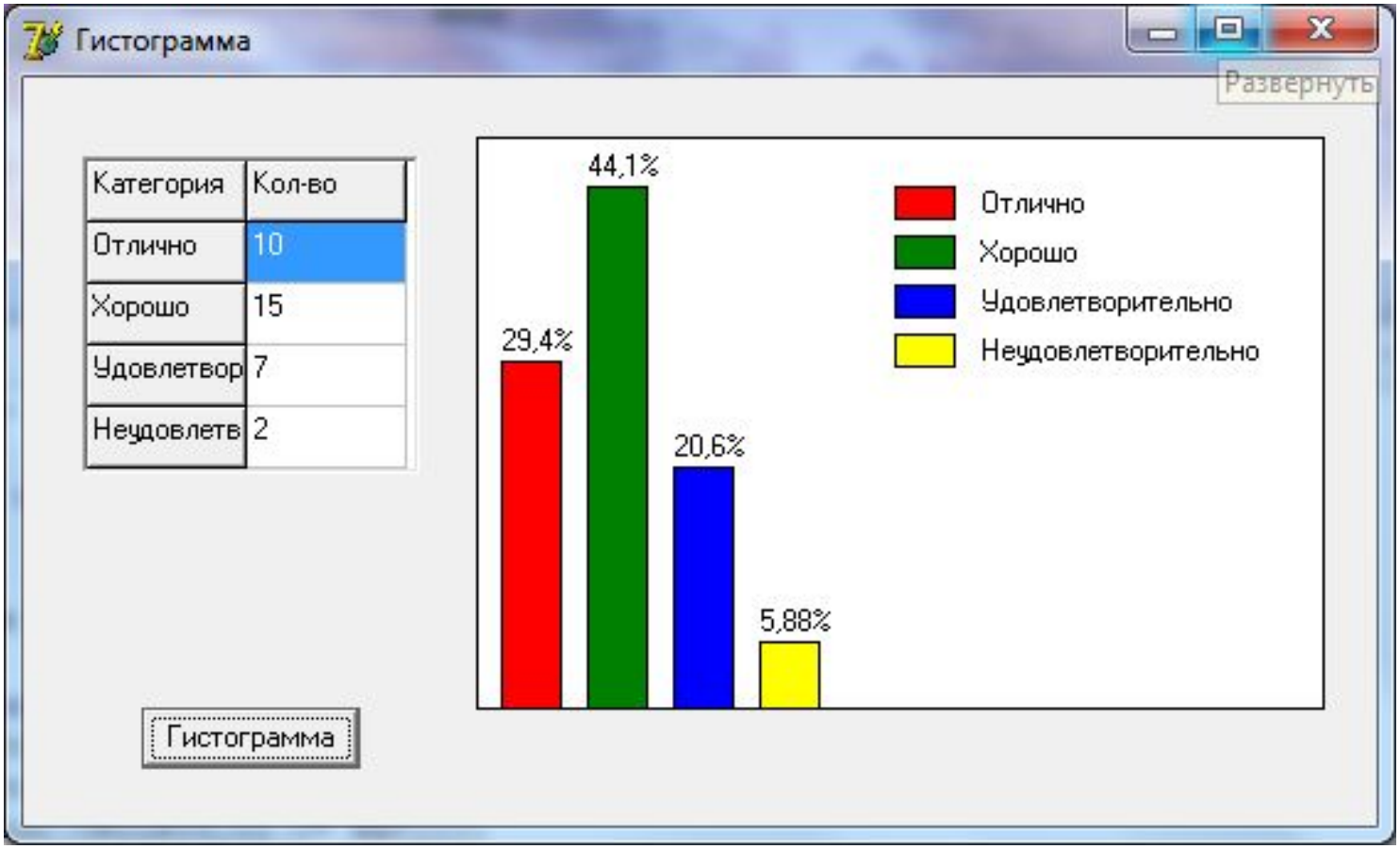
Построение диаграмм

Напишите программу, которая строит диаграмму, отображающую значения четырех категорий, например, результаты экзамена (категории: отлично, хорошо, удовлетворительно, неудовлетворительно). Для ввода исходных данных используйте компонент StringGrid



```
var
  Form1: TForm1;
  var FormatSettings:TFormatSettings;
  implementation
{$R *.dfm}
const  NR = 4; // кол-во строк в таблице
var kat: array[1..NR] of string =
('Отлично','Хорошо','Удовлетворительно','Неудовлетворительно');
 n: array[1..NR] of real; // значения категорий
 p: array[1..NR] of real; // процент категории в общей сумме
 h: array[1..NR] of integer; // высота столбиков диаграммы цвет столбиков диаграммы
 BarColor: array[1..4] of TColor = (clRed,clGreen,clBlue,clYellow);
// ввод и обработка если исходные данные введены, то Obr = TRUE
function Obr : boolean;
var sum: real; // сумма категорий
 m: integer; // номер категории, имеющей максимальное значение
 i: integer;
begin obr := FALSE; // пусть исх. данные не введены
 // скопируем содержимое второго столбца в массив исходных данных
 for i:=1 to NR do
 // здесь возможно исключение преобразования, если пользователь не ввел данные
 begin
 try
 n[i] := StrToFloat(Form1.StringGrid1.Cells[1,i]);
 except
 on EConvertError do
 begin
 ShowMessage('Надо ввести данные во все' + #13 + 'ячейки второй колонки.');
 exit;
 end;
 end;
 end; // вычислим сумму категорий (эл-тов второго столбца)
 sum := 0;
 for i:=1 to NR do
```

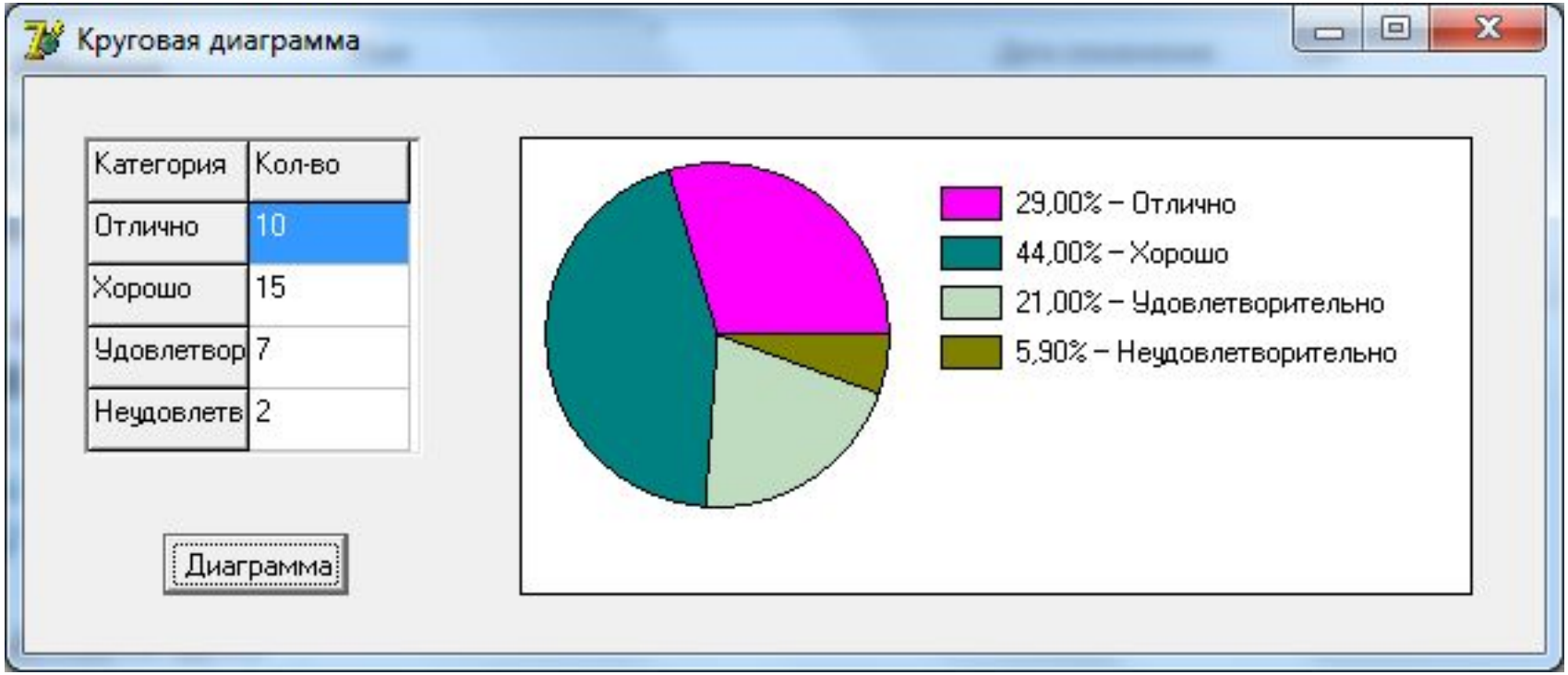
```
 sum := sum + n[i];  // вычислим процент каждой категории
 for i:=1 to NR do
 p[i] := n[i] / sum; // определим категорию с максимальным значением
 m := 1;
 for i := 2 to NR do
 if n[i] > n[m] then m:=i; // пусть максимальному значению соответствует
 // столбик высотой в Image1.Height-20 пикселов вычислим высоту остальных столбиков
 for i :=1 to NR do
 h[i] := Round((Form1.Image1.Height - 20) * n[i]/n[m]);
 // все готово можно строить диаграмму
 obr := TRUE;
end; // диаграмма
procedure diagr;
const WR = 25; // ширина столбика
 DR = 10; // расстояние между столбиками
var  x,y: integer; // левый нижний угол столбика
 i: integer;
  begin
 with Form1.Image1 do
 begin
 x:=10;
 y:=Height;
 Canvas.Brush.Color := clWindow;
 Canvas.Rectangle(0,0,Width,Height);  // *** рисуем столбики ***
 for i:=1 to 4 do
 begin
 Canvas.Brush.Color := BarColor[i]; // цвет столбика
 Canvas.Rectangle(x,y,x+WR,y-h[i]); // столбик
 Canvas.Brush.Color := clWindow; // чтобы область за текстом не была окрашена
 Canvas.TextOut(x,y-h[i]-15,
 FloatToStrF(p[i]*100,ffGeneral,3,2)+'%');
 x := x + WR + DR;
 end; // легенда здесь x — координата левой границы последнего столбика
 x := x + 20;
 y:=20; // 20 пикселов от верхнего края Image1
```

```
 for i:=1 to 4 do
 begin
 Canvas.Brush.Color := BarColor[i]; // цвет прямоугольника легенды
 Canvas.Rectangle(x,y,x+25,y+14); // прямоугольник легенды
 Canvas.Brush.Color := clWindow;
 Canvas.TextOut(x+WR+10,y,
 Form1.StringGrid1.Cells[0,i]);
 y := y + 20;
 end;
 end; // with Form1.Image1
end;
procedure TForm1.FormCreate(Sender: TObject);
begin // определим заголовки колонок
 StringGrid1.Cells[0,0] := 'Категория';
 StringGrid1.Cells[1,0] := 'Кол-во';
 StringGrid1.Cells[0,1] := kat[1];
 StringGrid1.Cells[0,2] := kat[2];
 StringGrid1.Cells[0,3] := kat[3];
 StringGrid1.Cells[0,4] := kat[4];
 StringGrid1.Cells[1,1] := '10';
 StringGrid1.Cells[1,2] := '15';
 StringGrid1.Cells[1,3] := '7';
 StringGrid1.Cells[1,4] := '2';
end;
procedure TForm1.StringGrid1KeyPress(Sender: TObject; var Key: Char);
begin // Col, Row — номер колонки и строки, в которой находится курсор.
 // ColCount и RowCount — кол-во колонок и строк
 if Key = #13 then
 begin  // нажата клавиша <Enter>
 if StringGrid1.Col < StringGrid1.ColCount - 1
 then  // ячейка не в последнем столбце
 StringGrid1.Col := StringGrid1.Col + 1 // к след. столбцу
 else // ячейка в последнем столбце
 if ( StringGrid1.Row < StringGrid1.RowCount - 1) then
 begin // в первый столбец следующей строки
 StringGrid1.Col :=0;
```

```
StringGrid1.Row := StringGrid1.Row +1;
end
else Button1.SetFocus;
exit;
end; // во вторую колонку разрешается вводить только числа
if StringGrid1.Col = 1 then // клавиша нажата в ячейке второй колонки
 case Key of '0'..'9', #8:;
'.',',':
  begin
 Key := FormatSettings.DecimalSeparator;
if Pos(FormatSettings.DecimalSeparator, StringGrid1.Cells[StringGrid1.Row,
StringGrid1.Col]) <> 0
then Key := Char(0);
end;
else Key := Char(0);
end;
end;
procedure TForm1.Button1Click(Sender: TObject);
begin
if Obr // исходные данные введены
then diagr; // строим диаграмму
end;

end.
```

Напишите программу, которая отражает ряд данных, например, результаты сдачи экзамена в виде круговой диаграммы



```
const NR = 4; // кол-во строк в таблице
var kat: array[1..NR] of string = ('Отлично','Хорошо','Удовлетворительно','Неудовлетворительно');
n: array[1..NR] of real; // значения категорий
p: array[1..NR] of real; // процент категории в общей сумме цвет секторов диаграммы
BarColor: array[1..4] of TColor = (clFuchsia,clTeal,clMoneyGreen,clOlive);
// ввод и обработка если исх. данные введены, то Obr = TRUE
function Obr : boolean;
var sum: real; // сумма категорий
m: integer; // номер категории, имеющей максимальное значение
i: integer;
begin
obr := FALSE; // пусть исх. данные не введены скопируем содержимое второго столбца
// в массив исходных данных
for i:=1 to NR do // здесь возможно исключение (ошибка) преобразования если пользователь не ввел данные
begin
 try
 n[i] := StrToFloat(Form1.StringGrid1.Cells[1,i]);
 except
 on EConvertError do
begin
 ShowMessage('Надо ввести данные во все' + #13 + 'ячейки второй колонки.');
 exit;
 end;
 end;
end; // вычислим сумму категорий (эл-тов второго столбца)
sum := 0;
for i:=1 to NR do
 sum := sum + n[i]; // вычислим процент каждой категории
for i:=1 to NR do
 p[i] := n[i] / sum; // все готово можно строить диаграмму
obr := TRUE;
end; // круговая диаграмма
procedure cdiagr;
var x1,y1,x2,y2 : integer; // круг (эллипс)
x0,y0: integer; // центр круга (эллипса)
```

```
x3,y3,x4,y4 : integer; // линии, ограничивающие сектор
a1,a2: integer; // угол начала и конца сектора
x,y: integer; // координаты области вывода легенды (левый верхний угол)
i: integer;
r: integer;
st: string;
begin
x1:= 10; y1:=10;
x2:= 150; y2 :=150;
x0 := x1 + (x2-x1) div 2;
y0 := y1 + (y2-y1) div 2;
r := (x2-x1) div 2;
with Form1.Image1 do
begin
 Canvas.Brush.Color := clWindow;
 Canvas.Rectangle(0,0,Width,Height);
 a1 := 1; // строим от оси OX
 x3 := x2;
 y3 := y1 + r;
 for i:=1 to 4 do
 begin
 if i < 4 then
 a2 := a1 + Round(360 * P[i])
 else
 a2 := 360;
 x4:= x0 + Round( r* cos(a2*PI/180));
 y4:= y0 - Round( r* sin(a2*PI/180));
 Canvas.Brush.Color := BarColor[i];
 Canvas.Pie(x1,y1,x2,y2,x3,y3,x4,y4); // следующий сектор — от конца текущего
 a1 := a2;
 x3:=x4;
 y3:=y4;
 end; // Легенда. Здесь x — коорд. левого угла прямоугольника легенды
 x := x2 + 20;
 y := 20; // 20 пикселов от верхнего края Image1
```

```
 for i:=1 to 4 do
 begin
 Canvas.Brush.Color := BarColor[i];// цвет прямоугольника легенды
 Canvas.Rectangle(x,y,x+25,y+14); // прямоугольник легенды
 Canvas.Brush.Color := clWindow;
 st := FloatToStrF(p[i]*100,ffFixed,2,2) + '% —' + Form1.StringGrid1.Cells[0,i];
 Canvas.TextOut(x+30,y, st);
 y := y + 20;
 end;
end; // with Form1.Image1
end;
procedure TForm1.FormCreate(Sender: TObject);
begin // определим заголовки колонок
StringGrid1.Cells[0,0] := 'Категория';
StringGrid1.Cells[1,0] := 'Кол-во'; // и содержимое первого столбца
StringGrid1.Cells[0,1] := kat[1];
StringGrid1.Cells[0,2] := kat[2];
StringGrid1.Cells[0,3] := kat[3];
StringGrid1.Cells[0,4] := kat[4];
StringGrid1.Cells[1,1] := '10';
StringGrid1.Cells[1,2] := '15';
StringGrid1.Cells[1,3] := '7';
StringGrid1.Cells[1,4] := '2';
end;
procedure TForm1.StringGrid1KeyPress(Sender: TObject; var Key: Char);
begin
if Key = #13 then
begin // нажата клавиша <Enter>
if StringGrid1.Col < StringGrid1.ColCount - 1
 then  // ячейка не в последнем столбце
 StringGrid1.Col := StringGrid1.Col + 1 // к след. столбцу
 else // ячейка в последнем столбце
 if (StringGrid1.Row < StringGrid1.RowCount - 1) then
 begin // в первый столбец следующей строки
 StringGrid1.Col :=0;
```

```
StringGrid1.Row := StringGrid1.Row + 1;
end
else BitBtn1.SetFocus;
exit;
 end; // во вторую колонку разрешается вводить только числа
if StringGrid1.Col = 1 then // клавиша нажата в ячейке второй колонки
case Key of
'0'..'9',#8: ;
'.',',':
begin
Key := DecimalSeparator;
if Pos(DecimalSeparator,
StringGrid1.Cells[StringGrid1.Row,
StringGrid1.Col]) <> 0
then Key := Char(0);
end;
else Key := Char(0);
end;
end;
// щелчок на кнопке Построить
procedure TForm1.BitBtn1Click(Sender: TObject);
begin
if Obr // исх. данные введены
then cdiagr; // строим диаграмму
end;
```