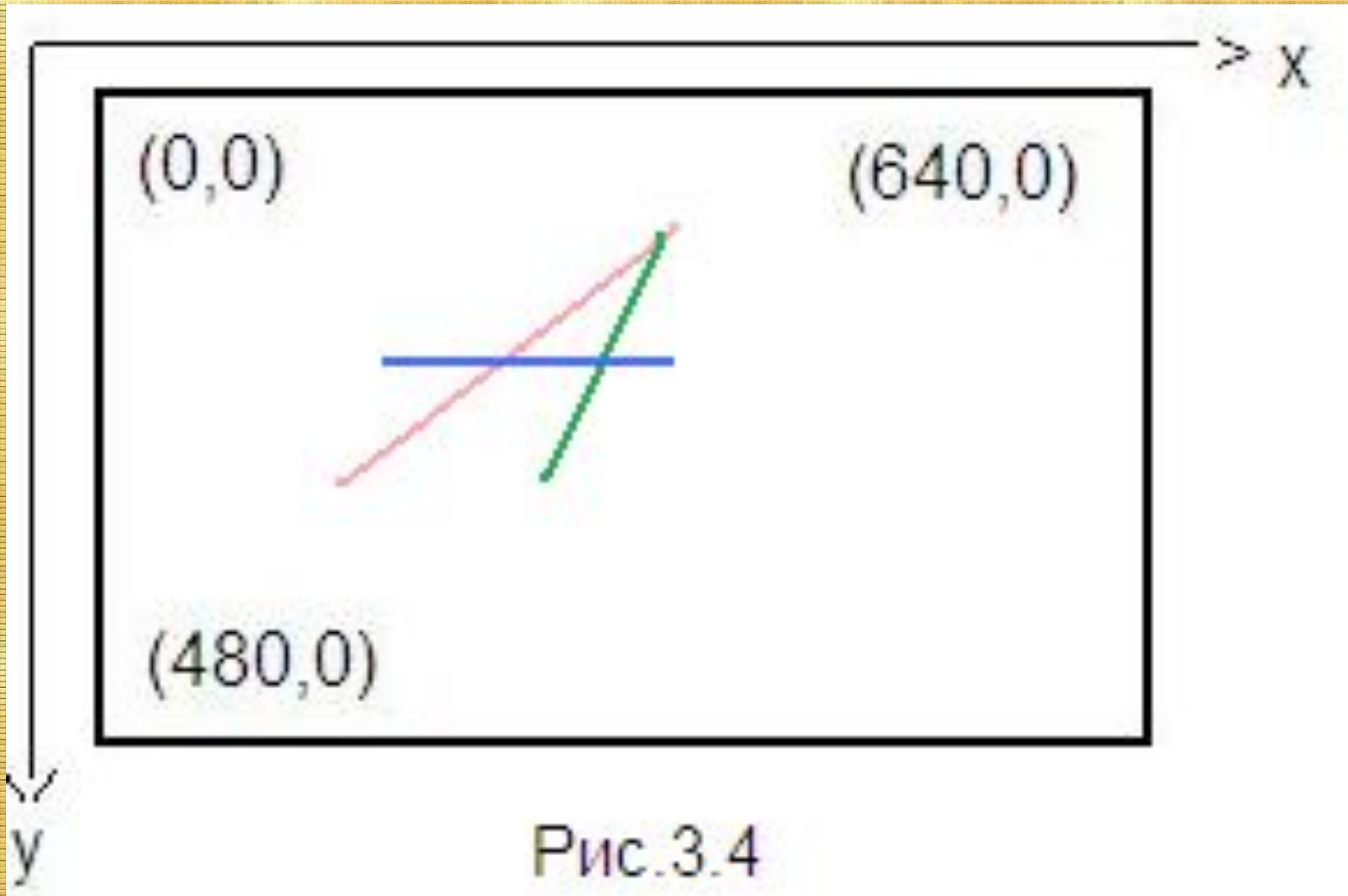


# Графические возможности языка программирования

Каждый пиксель (точка) имеет две координаты:  $x$  и  $y$



Геометрические размеры пикселя определяются *разрешением* монитора.

## Пример 1. Демонстрация подключения модуля GraphAbs

**program** точка;

**uses** graphabc;                    *{подключение модуля GraphAbs}*

**begin**

**setwindowsize(640,480);** *{устанавливает размеры графического окна}*

**setpixel(100,120,clBlack);** *{устанавливает черный цвет*

*пера и рисует точку с координатами (100,120)}*

**end.**

В данном примере мы познакомились с использованием команды `setpixel` в частном случае. В общем виде эта команда выглядит так:

**setpixel(x:integer,y:integer,c:color)** – рисует точку с координатами (x,y)

цветом c.

clBlack – черный

clWhite – белый

clGreen – зеленый

clBlue – синий

clYellow – желтый

clPurple – фиолетовый

clRed – красный

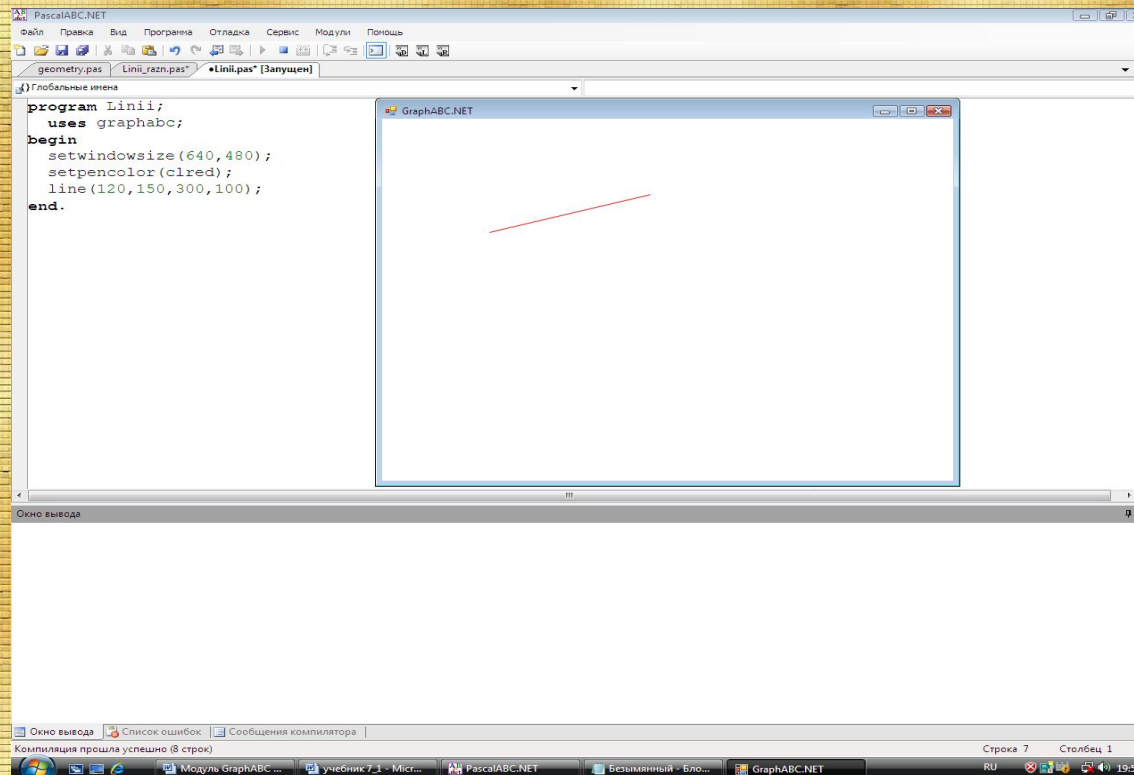
clBrown – коричневый

clSkyBlue – голубой

Программа рисования отрезка, соединяющего две точки с координатами (120,150) и (150,80) красным цветом пера, может выглядеть так:

Пример 2. Демонстрация рисования линии

```
program Lini;  
uses graphabc;  
begin  
  setwindowsize(640,480);  
  setpencolor(clred);      {устанавливает красный цвет пера}  
  line(120,150,300,100); {рисует отрезок от (120,150) до (300,100)}  
end.
```



### Пример 3. Демонстрация рисования линий разным цветом пера

**program** treugolnik;

**uses** graphabc;

**begin**

setwindowsize(640,480);

**setpenwidth(5);** *{устанавливает ширину текущего пера. В скобках указывается количество пикселей, образующих ширину линии}*

**setpencolor(clRed);** *{устанавливает красный цвет пера}*

**line(100,200,170,70);** *{рисует отрезок от точки с координатами (100,200) до точки с координатами (170,70)}*

**setpencolor(clGreen);** *{устанавливает зеленый цвет пера}*

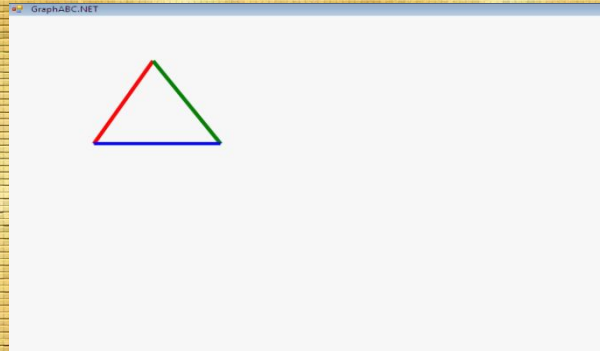
**line(170,70,250,200);** *{рисует отрезок от точки(170,70) до точки(250,200)}*

**setpencolor(clBlue);** *{устанавливает синий цвет пера}*

**line(250,200,100,200);** *{рисует отрезок от точки(250,200) до точки(100,200)}*

*{результат - треугольник со сторонами разных цветов}*

**end.**



### Важно помнить:

1. Рисуя линии, можно задавать ее размер (координатами ее концов), цвет, ширину (толщину) и стиль.
2. Для задания стиля линии в модуле GraphAbs существует процедура `SetPenStyle (Style)`, где `Style` - константы стилей пера (см. Приложение к главе 3).
3. Линия может быть сплошной, пунктирной, штрихпунктирной, штриховой. Замкнутые фигуры можно закрашивать.

## Пример 4. Демонстрация рисования замкнутых фигур с помощью линий и их закрашивание

```
Program treug_zakrash;
```

```
  uses graphabc;
```

```
begin
```

```
  setwindowsize(640,480);
```

```
  clearwindow(clWhite); {очищает графическое окно белым  
цветом}
```

```
  setpenwidth(3);    {устанавливает ширину текущего пера}
```

```
  setpenstyle(pssolid); {устанавливает стиль линии –  
сплошная линия}
```

```
  setpencolor(clgreen); {устанавливает зеленый цвет пера}
```

```
  line(100,200,170,70); {рисует линии зеленым цветом}
```

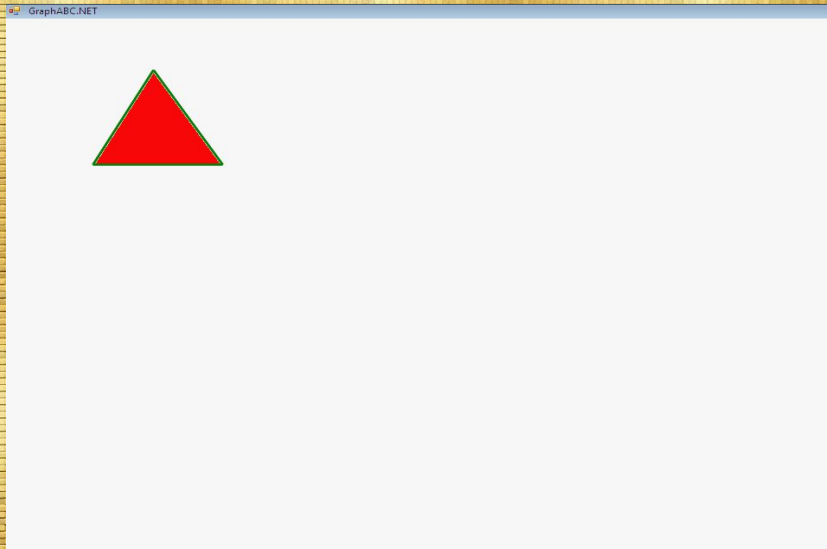
```
  line(170,70,250,200);
```

```
  line(250,200,100,200);
```

```
  floodfill(440,120,clred); {Закрашивает треугольник  
красным цветом}
```

```
end.
```

В результате выполнения программы на экране монитора в графическом окне появится треугольник, нарисованный зеленым и закрашенный красным цветом



Важно помнить:

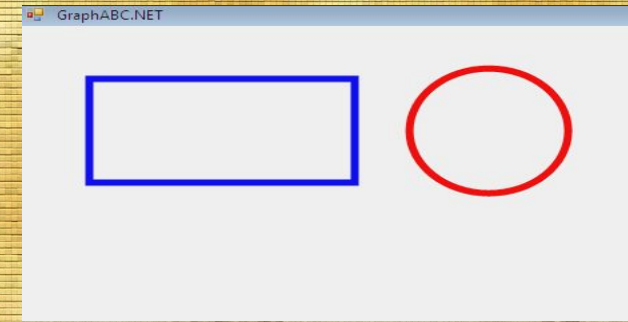
1. Закрашивать можно только замкнутые фигуры, контур которых нарисован одним цветом.
2. В процедуре заливки **floodfill(x,y,c)** указывается координата точки (x,y), которая обязательно должна попасть во внутреннюю область закрашиваемой фигуры.



Прямоугольники и окружности можно рисовать с помощью команд **rectangle(x1,y1,x2,y2)** и **circle(x,y,r)** соответственно. Как это можно сделать, рассмотрим на примере программы, которая рисует прямоугольник и окружность.

Пример 5. **Демонстрация рисования прямоугольника и окружности**

```
program gemetry;  
uses graphabc;  
begin  
  setwindow(640,480);  
  setpencolor(clBlue); {устанавливает голубой цвет пера  
                        для рисования контура прямоугольника}  
  setpenwidth(6);      {устанавливает ширину пера}  
  rectangle(50,50,250,150);{рисует прямоугольник, заданный  
                           координатами противоположных вершин}  
  setpencolor(clRed);  {устанавливает красный цвет пера  
                        для рисования контура окружности}  
  circle(350,100,60); {рисует окружность с центром в точке  
                      с координатами (350,100) и радиусом 60}  
end.
```



# Демонстрация закрашивания прямоугольника и окружности и их надписи

```
program gemetry3;
```

```
uses graphabc;
```

```
begin
```

```
  setwindow(640,480);
```

```
  clearwindow(clYellow); {устанавливает желтый цвет фона}
```

```
  setpencolor(clteal); {устанавливает сине-зеленый цвет пера}
```

```
  setpenwidth(5); {устанавливает ширину линии}
```

```
  setbrushcolor(clolive); {устанавливает оливковый цвет кисти}
```

```
  rectangle(100,100,300,200); {рисует закрашенный оливковым цветом прямоугольник}
```

```
  setbrushcolor(clblue); {устанавливает синий цвет кисти}
```

```
  circle(400,150,50); {рисует закрашенный синим цветом круг}
```

```
  setfontstyle(fsbold); {устанавливает стиль шрифта}
```

```
  setfontsize(15); {устанавливает размер шрифта}
```

```
  setbrushcolor(clwhite); {устанавливает белый цвет кисти}
```

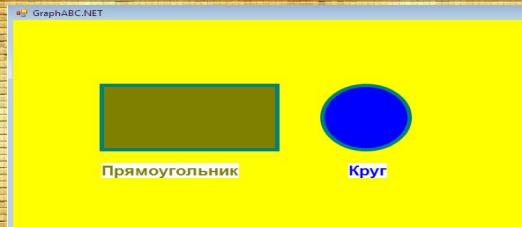
```
  setfontcolor(clolive); {устанавливает оливковый цвет шрифта}
```

```
  textout(100,220,'Прямоугольник'); {делает надпись}
```

```
  setfontcolor(clblue); {устанавливает синий цвет шрифта}
```

```
  textout(380,220,'Круг'); {делает надпись}
```

```
end.
```



## Графические процедуры АВС Паскаля:

1. **SetPixel(x,y,color: integer);** - закрашивает один пиксель с координатами (x,y) цветом color.
2. **Line(x1,y1,x2,y2: integer);** - рисует отрезок от точки (x1,y1) и до точки (x2,y2).
3. **Circle(x,y,r: integer);** - рисует окружность с центром в точке (x,y) и радиусом r.
4. **Rectangle(x1,y1,x2,y2: integer);** -рисует прямоугольник, заданный координатами противоположных вершин (x1,y1) и (x2,y2).
5. **TextOut(x,y: integer; s: string);** - выводит строку s в позицию (x,y) (точка (x,y) задает верхний левый угол прямоугольника, который будет содержать текст из строки s).
6. **FloodFill(x,y,color: integer);** - заливает область одного цвета цветом color, начиная с точки (x,y).
7. **FillRect(x1,y1,x2,y2: integer);** - заливает прямоугольник, заданный координатами противоположных вершин (x1,y1) и (x2,y2), цветом текущей кисти.

Что будет выведено на экран монитора в результате выполнения данных программ?

```
program grafika1;  
uses graphabc;  
begin  
  setpenwidth(10);  
  setpencolor(clred);  
  Line(100,100,270,90);  
End.
```

```
Program grafika2;  
uses graphabc;  
begin  
  setpenwidth(8);  
  setpencolor(clblue);  
  circle (200,150,50);  
End.
```

## Перепишите в тетрадь:

**Тема:** Графические возможности языка программирования.

1. **Line(x1,y1,x2,y2);** - отрезок от (x1,y1) до (x2,y2).
2. **Circle(x, y, r);** - окружность с центром в (x, y) и радиусом r.
3. **Rectangle(x1,y1,x2,y2);** - прямоугольник, заданный координатами противоположных вершин (x1,y1) и (x2,y2).
4. **FloodFill(x, y, color);** - заливает область одного цвета цветом color, начиная с точки (x, y).