

ГРАФИЧЕСКИЙ РЕЖИМ

Turbo Pascal 7.0

Графические операции

- Система координат
- Инициализация
- Изображение точек
- Изображение линий
- Закрашенные фигуры
- Вывод текста



SetLineStyle(s, p, t)

Стиль

Шаблон

Толщина

0	_____
1	_____
2	_____
3	_____
4	Определяется шаблоном

1	Тонкая
3	Толстая

Цвет линий и текста

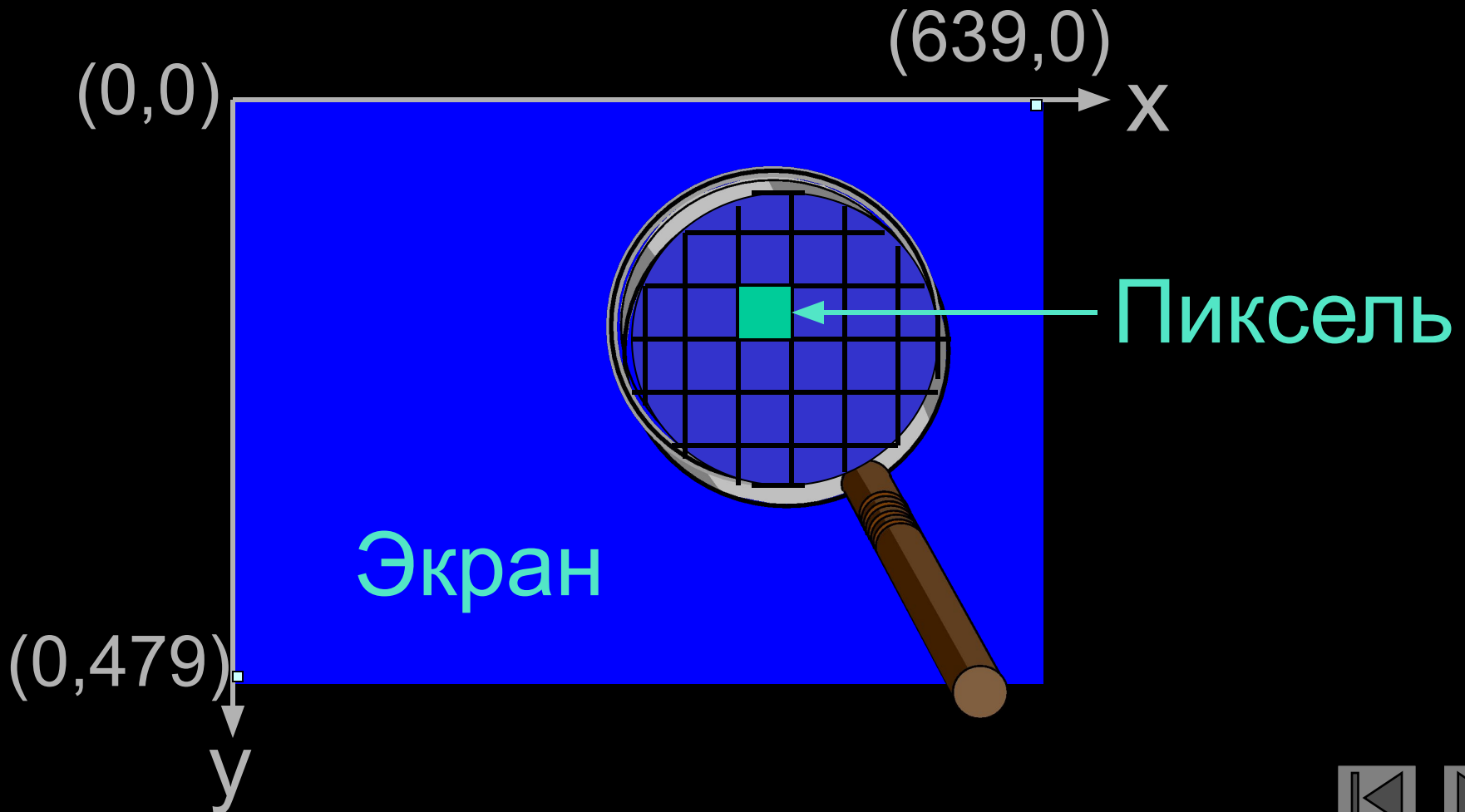
SetColor(c)

Цвет текста



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Экранная система координат



Инициализация (включение) графического режима

InitGraph (gd, gm, p)

Графический драйвер (gd=9) ↑
Режим (разрешение) ↓

Путь к драйверу ↑
↓

gm=0 - низкое (640x200)
gm=1 - среднее (640x350)
gm=2 - высокое (640x480)

'c:\trans\bp\bgi'



Выключение графического режима

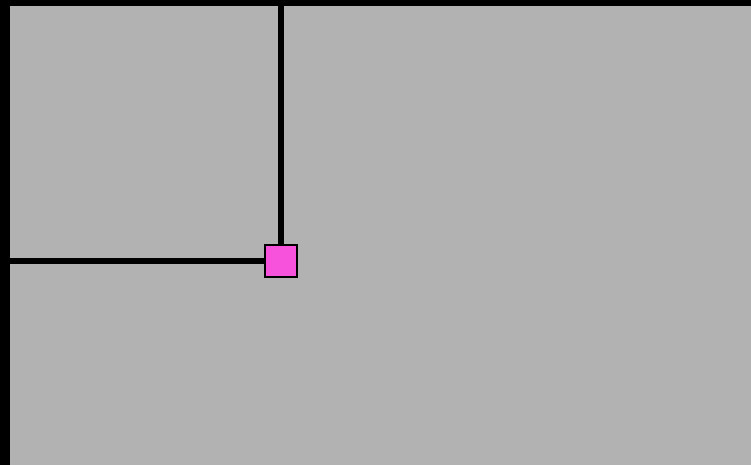
CloseGraph



Изображение точек

PutPixel (x, y, c);

Координаты 
Цвет 



Нарисовать точку зеленого цвета в центре экрана

```
Program Primer;  
Uses Graph;  
Var   Gd, Gm : Integer;  
Begin  
    Gd:=9; Gm:=2;  
    InitGraph(Gd,Gm,'c:\trans\bp\bgi');  
    PutPixel(320,240,2);  
    Readln;  
    CloseGraph;  
End.
```



Изображение линий

- Стиль и цвет линии
- Прямые линии
- Контур прямоугольника
- Окружность
- Дуга окружности
- Эллипс (дуга эллипса)

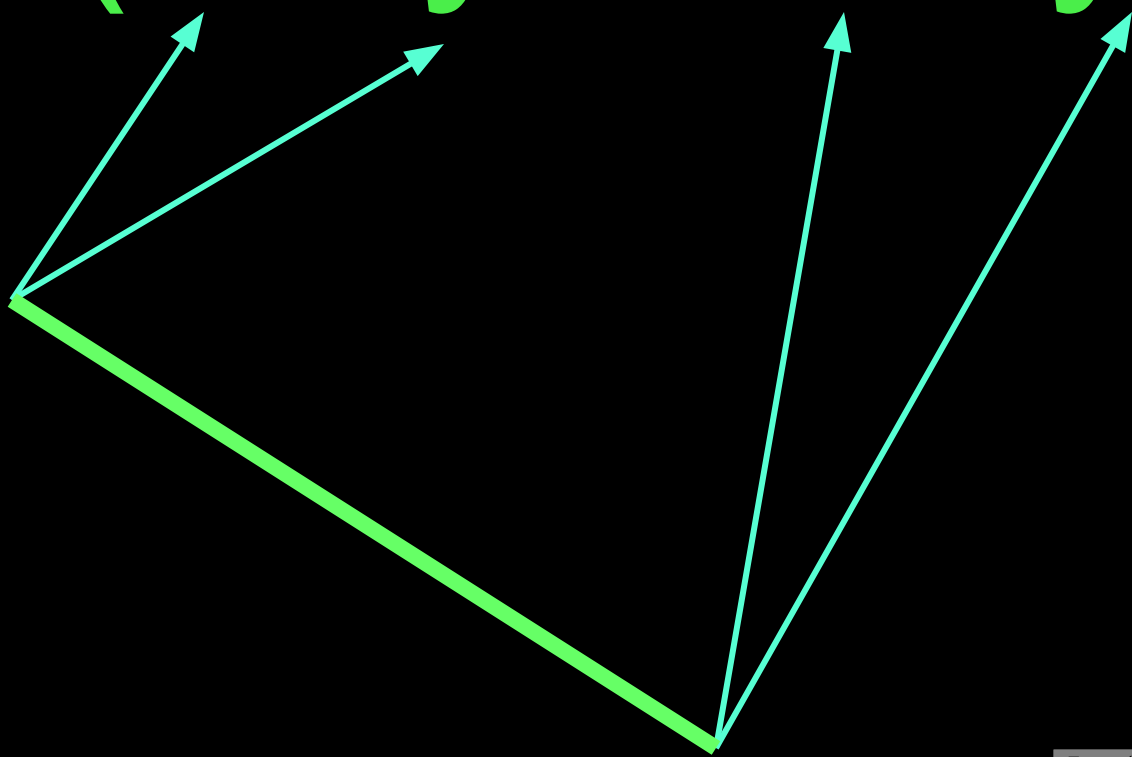
SetColor(c)

Цвет линии



Отрезок прямой линии

Line (x_1, y_1, x_2, y_2)



Отрезок до точки

LineTo (x,y)

Текущий указатель
до исполнения
команды

Текущий
указатель
после
исполнения
команды



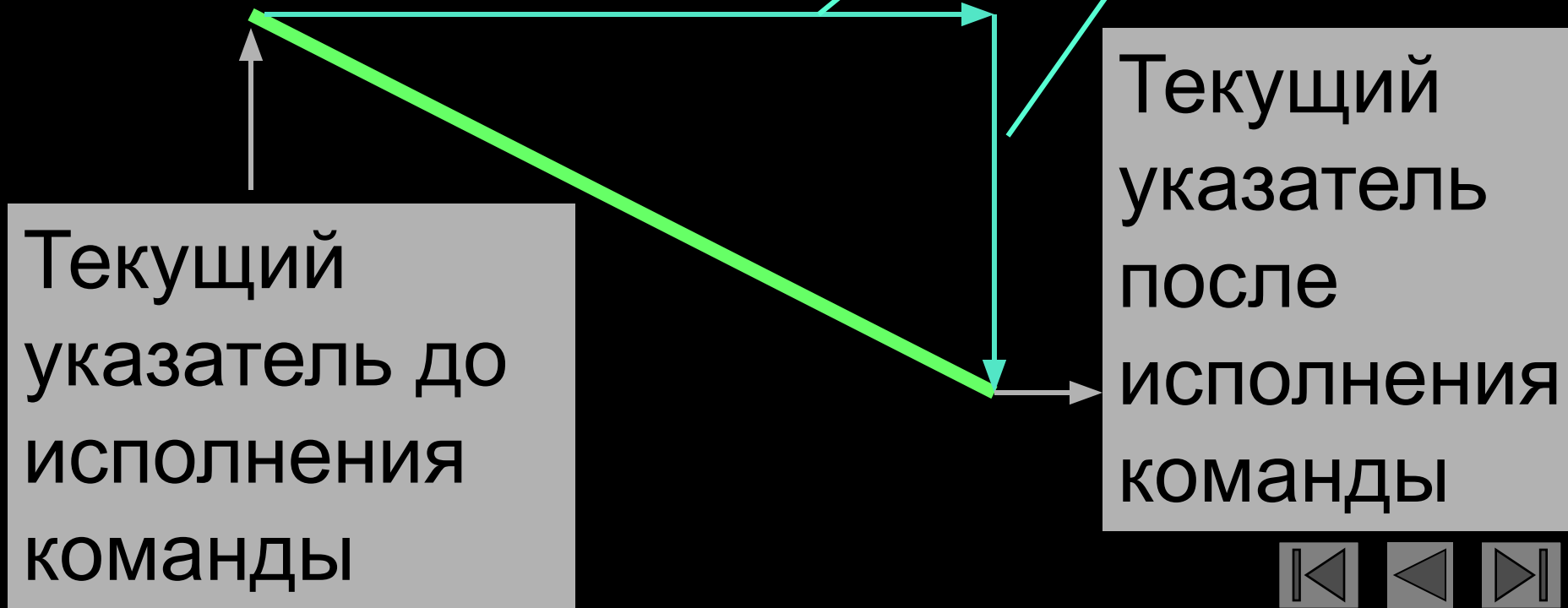
*Перемещение
текущего указателя
в точку с координатами (x,y)*

MoveTo (x,y)



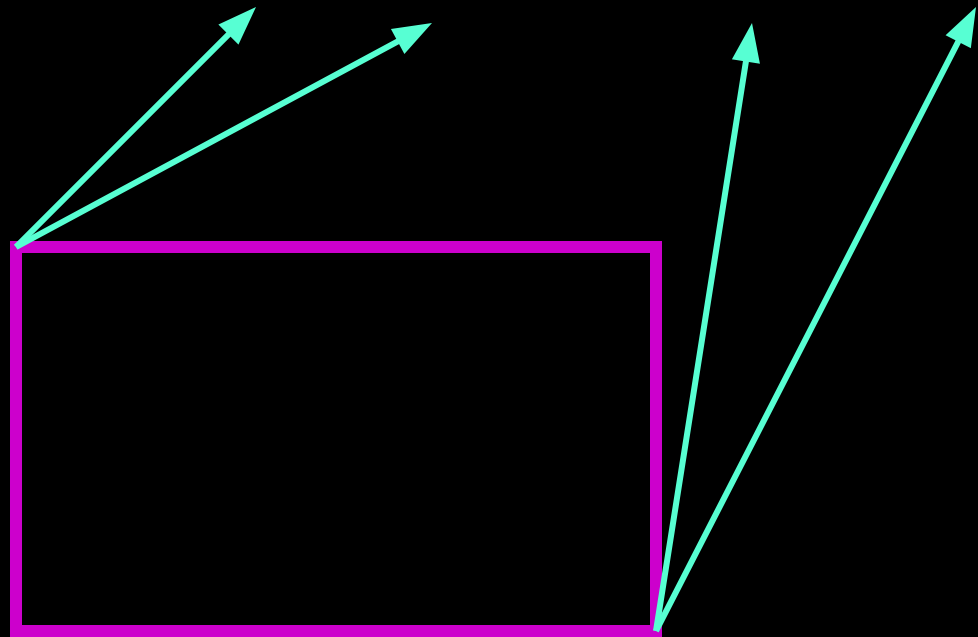
Отрезок (вектор)

LineRel (dx, dy)



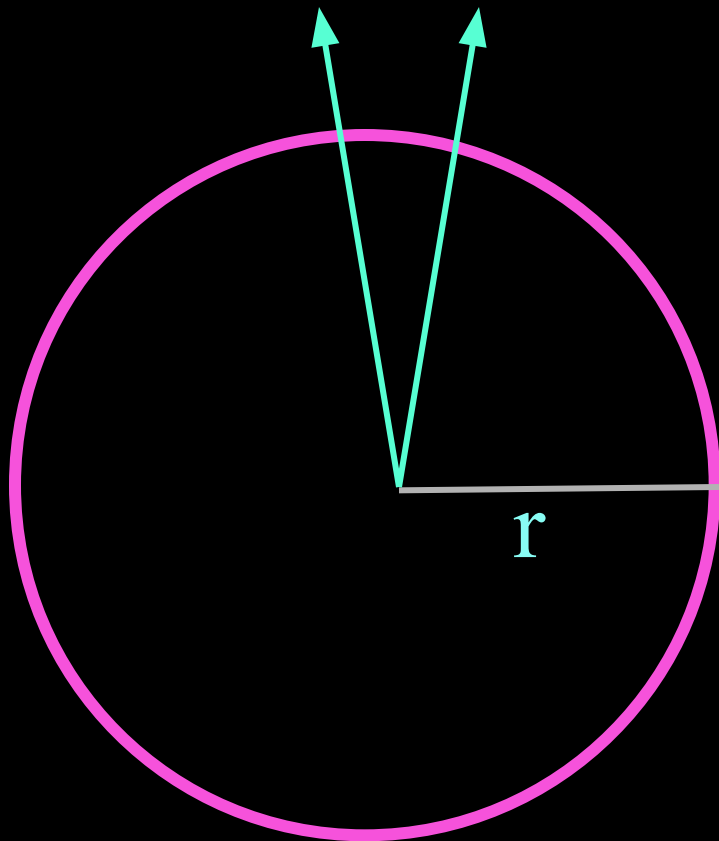
Контур прямоугольника

Rectangle(x1, y1, x2, y2)



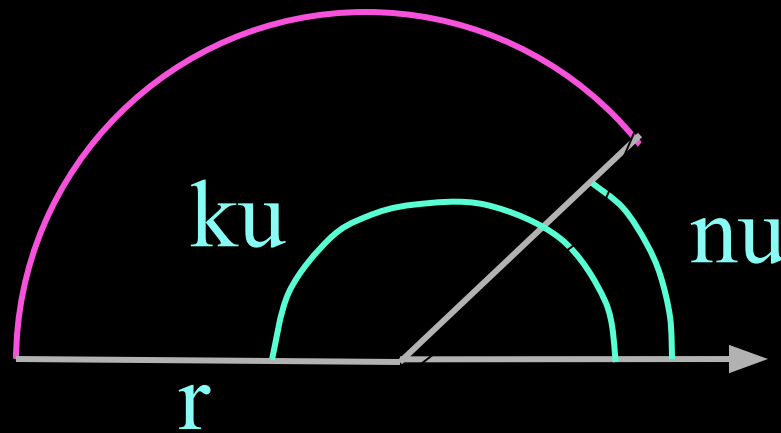
Окружность

Circle (x, y, r)



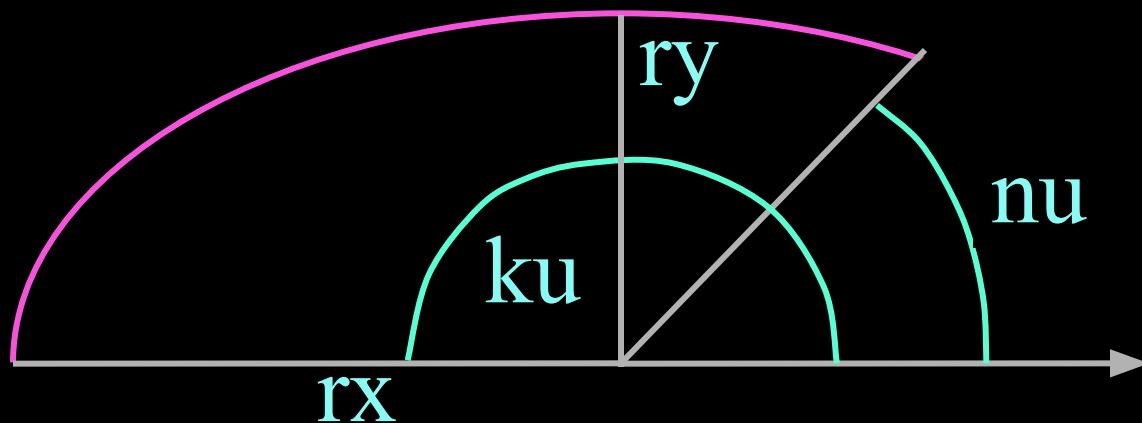
Дуга окружности

Arc (x,y,nu,ku,r)



Дуга эллипса

Ellipse(x, y, nu, ku, rx, ry)



Закрашенные фигуры

- Стиль заполнения фигуры
- Прямоугольник
- Параллелепипед
- Закрашенный эллипс
- Сектор круга
- Сектор эллипса
- Закраска произвольной фигуры



Стиль заполнения фигуры

SetFillStyle (s, c)

Стиль

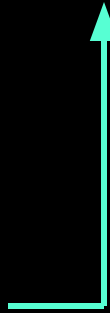
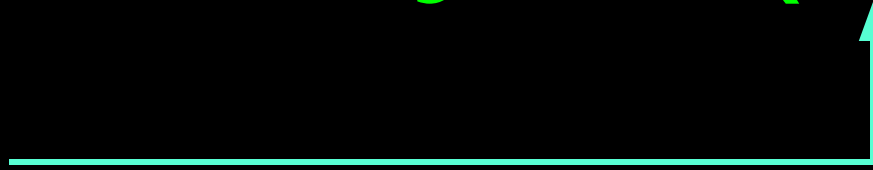


0 ... 12

Активный цвет



0 ... 15



Виды стилей заполнения:

- 0 - сплошная заливка фоновым цветом
- 1 - сплошная заливка активным цветом
- 9 - заливка пикселей в шахматном порядке
- 12 - пользовательский стиль

2

3

4

5

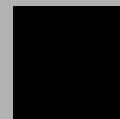
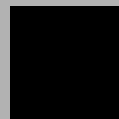
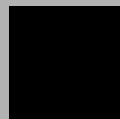
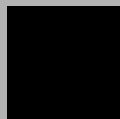
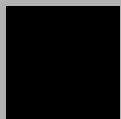
6

7

8

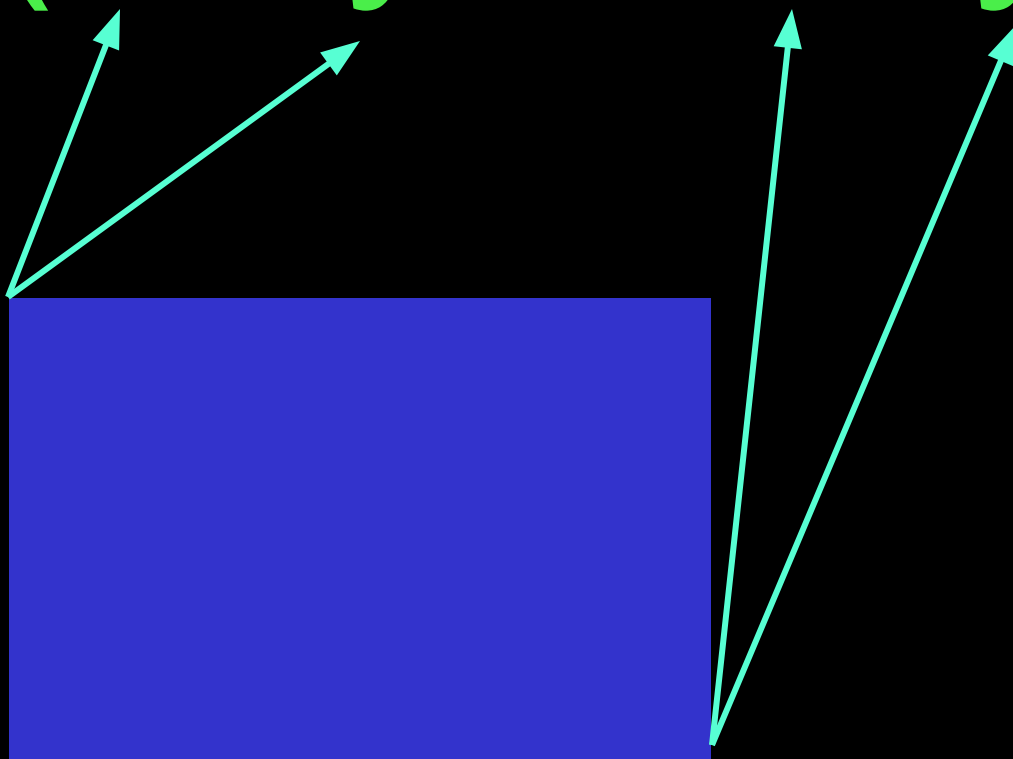
10

11



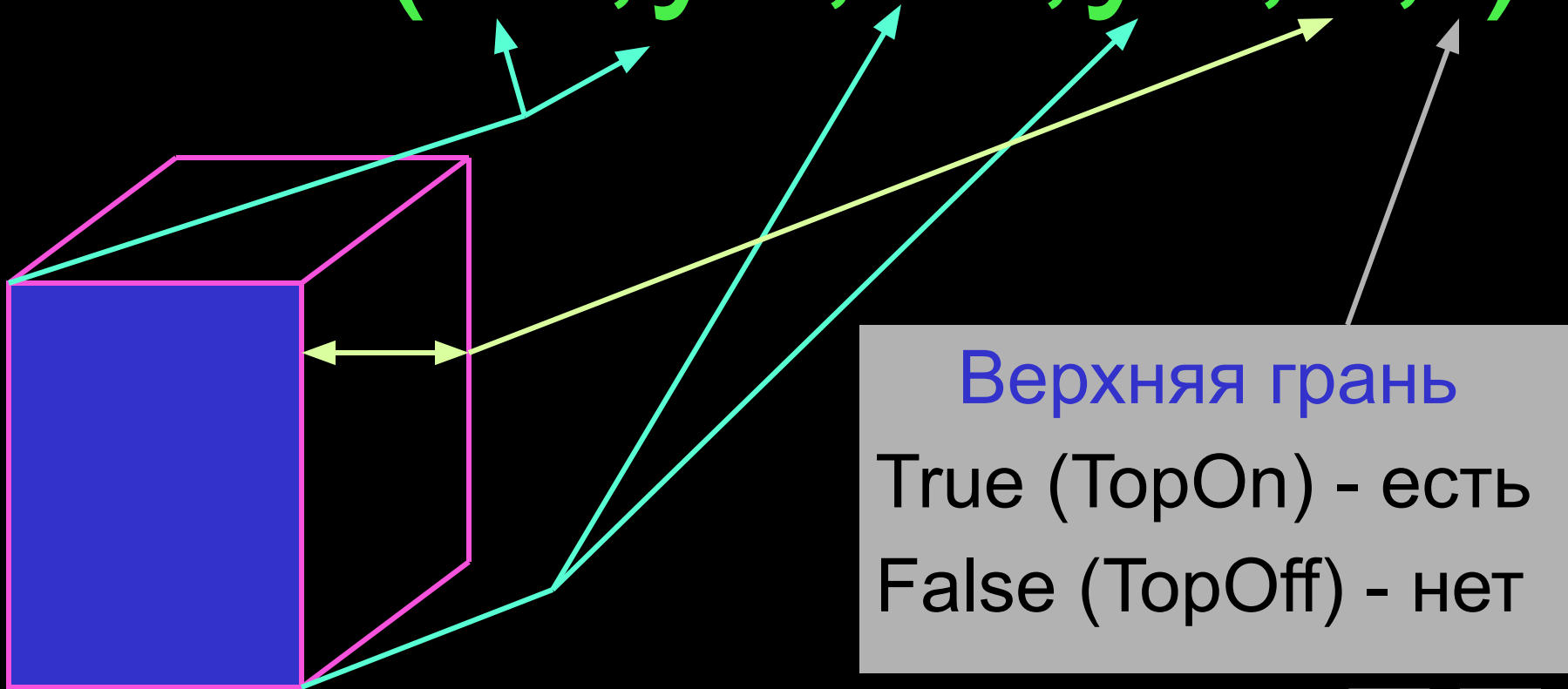
Закрашенный прямоугольник

Var (x1, y1, x2, y2)



Параллелепипед

Bar3d(x1,y1,x2,y2,d,t)



Верхняя грань

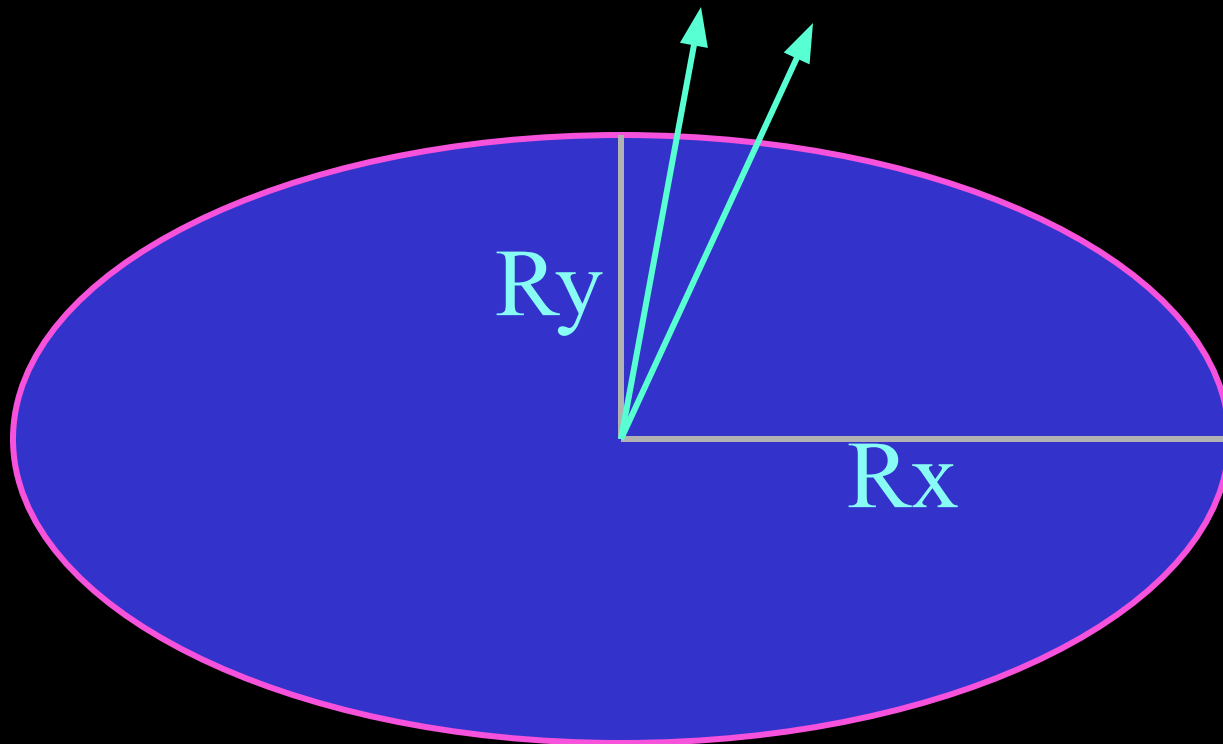
True (TopOn) - есть

False (TopOff) - нет



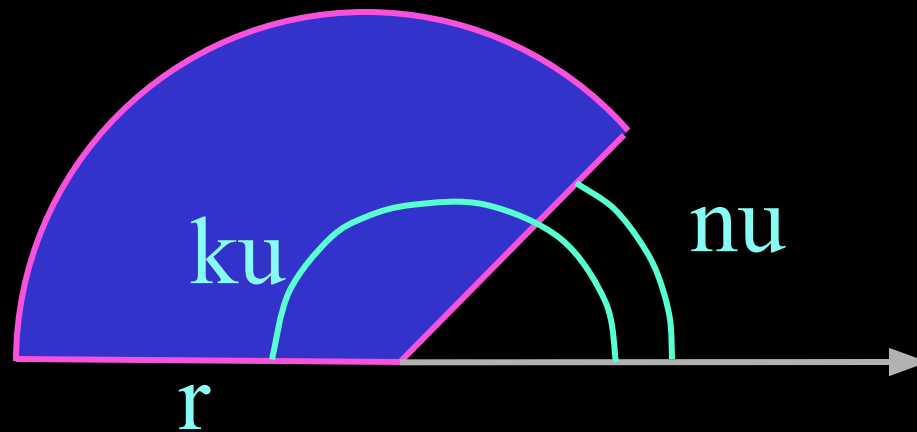
Закрашенный эллипс

FillEllipse (x,y,Rx,Ry)



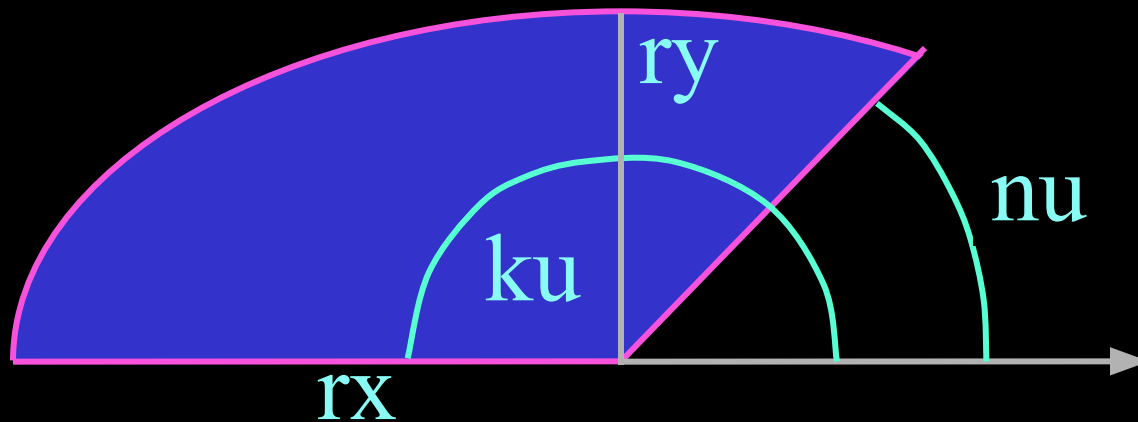
Закрашенный сектор круга

PieSlice (x,y,nu,ku,r)



Закрашенный сектор эллипса

Sector(x, y, nu, ku, rx, ry)

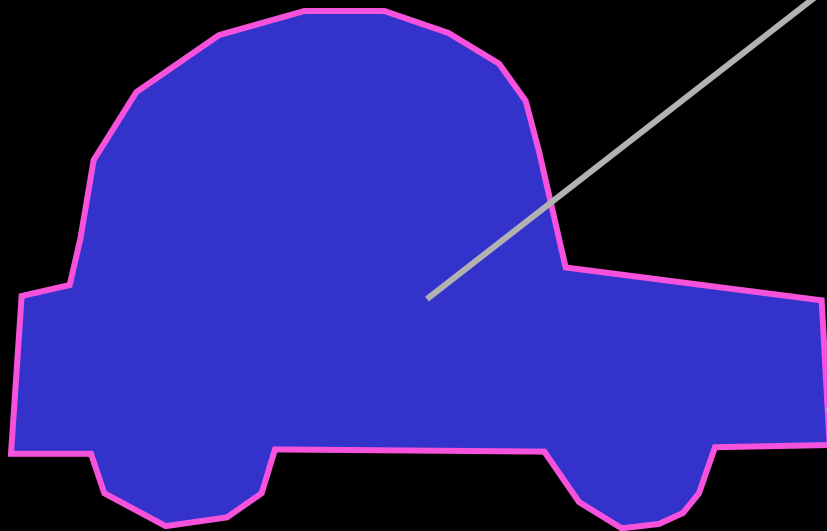


Закраска произвольной фигуры

FloodFill(x, y, b)

Координаты внутренней точки

Цвет границы



Вывод текста в графике

- **Стиль и цвет текста**
- Вывод текста
- Выравнивание текста
- Преобразование чисел в текстовую форму
- Высота и ширина строки

Стиль текста

SetTextStyle(s, d, r)

Стиль

Направление

Размер

0 - побитовый
1 - тройной
2 - малый
3 - гротесковый
4 - готический

0 - горизонтально (→)
1 - вертикально (↑)

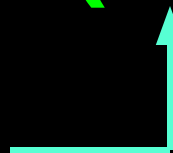
1..10



Цвет текста

SetColor(c)

Цвет текста



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Вывод текста

OutTextXY(x, y, s)

Координаты 

Текст 

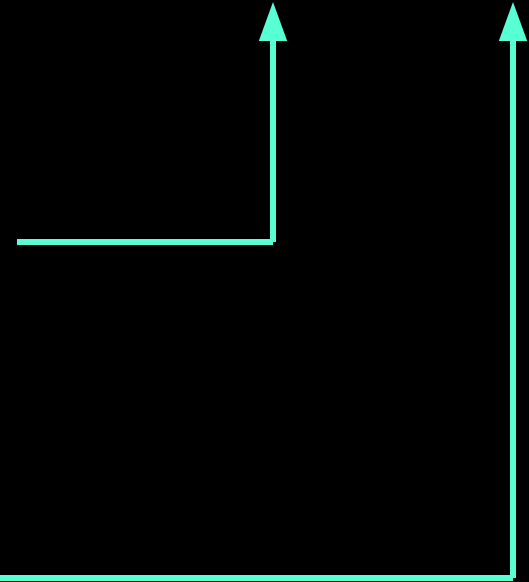
↓
Тип - string

Выравнивание текста

SetTextJustify(h, v)

Выравнивание
по горизонтали

Выравнивание
по вертикали



Выравнивание текста относительно точки привязки

• Текст

$h=0$

• Текст

$h=1$

• Текст

$h=2$

• Текст

$v=0$

• Текст

$v=1$

• Текст

$v=2$

Преобразование чисел в текст

`Str(x, s)`

Исходное число

Строка (текст)

Тип - string

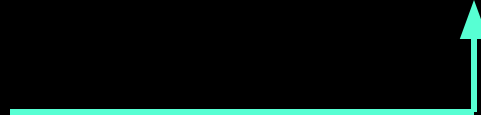


Ширина строки

Функция

TextWidth (S)

Строка



Ширина в
пикселях

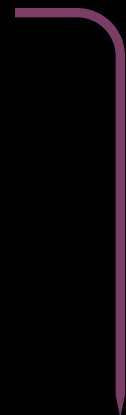


Высота строки

Функция

TextHeight(S)

Строка



Высота в
пикселях

