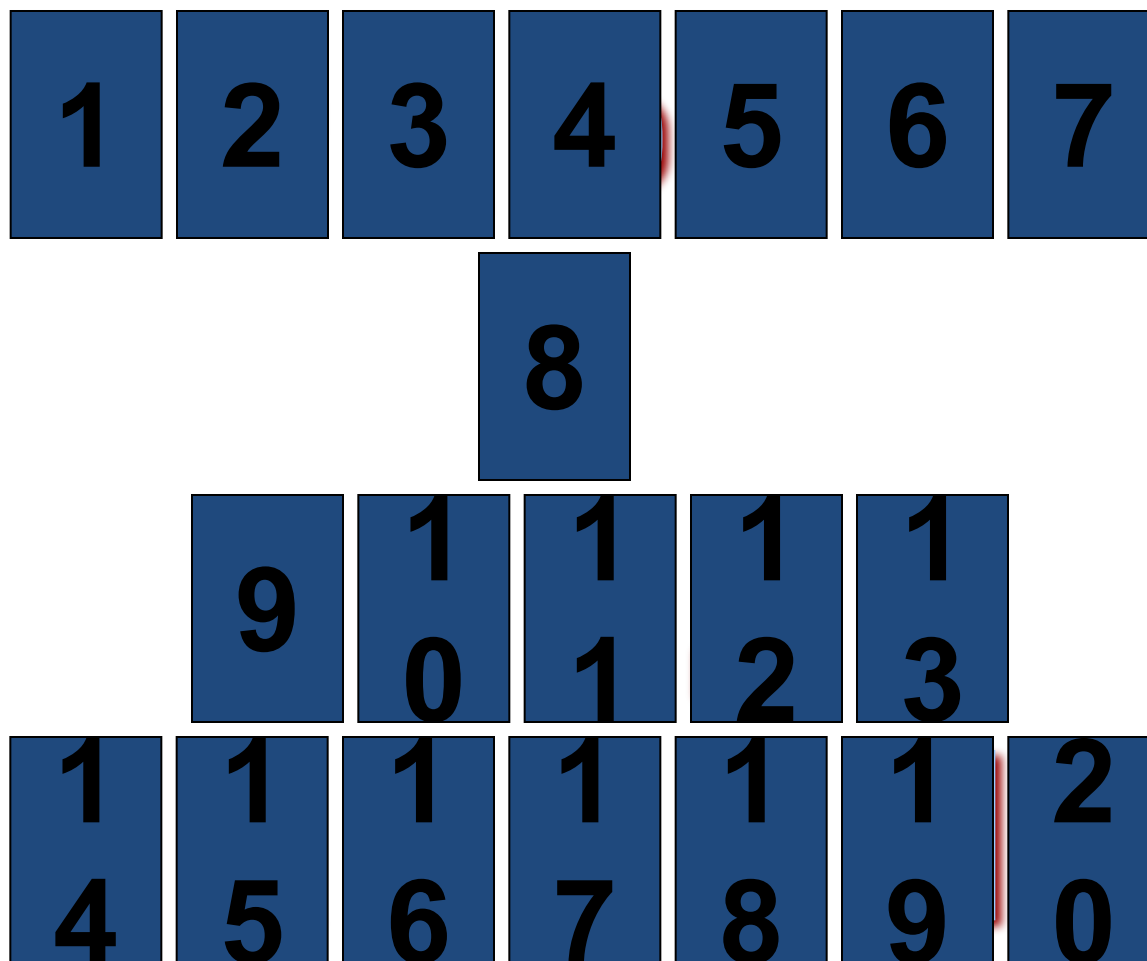
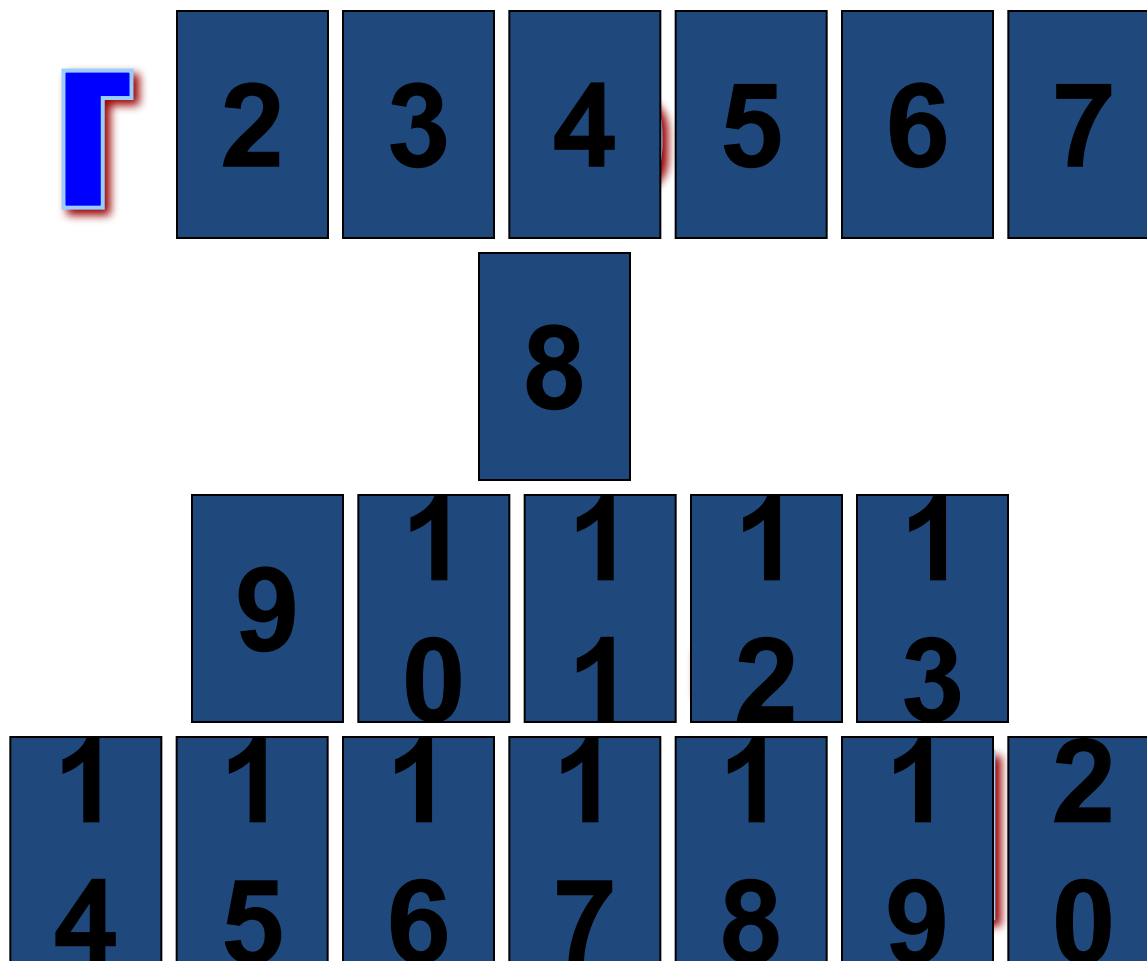


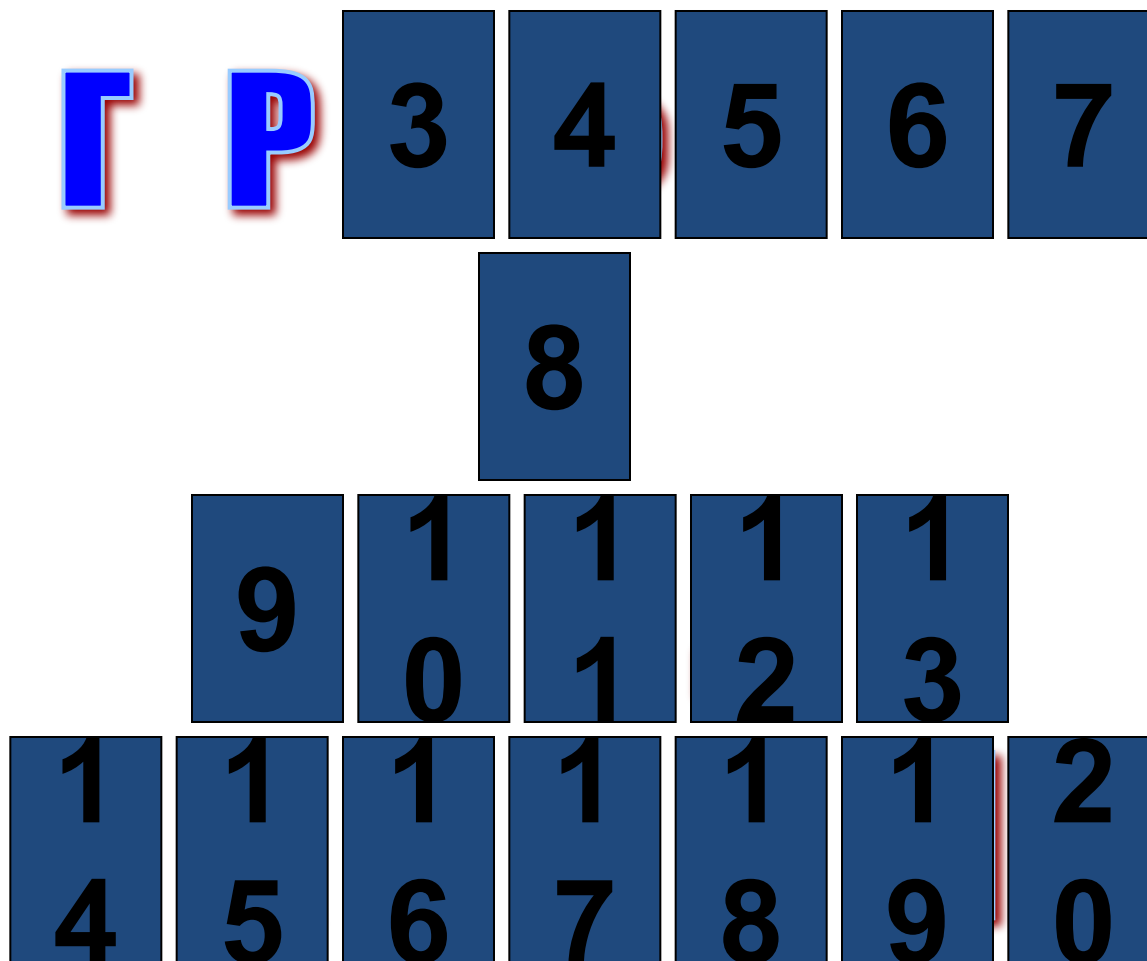
# Тема урока:



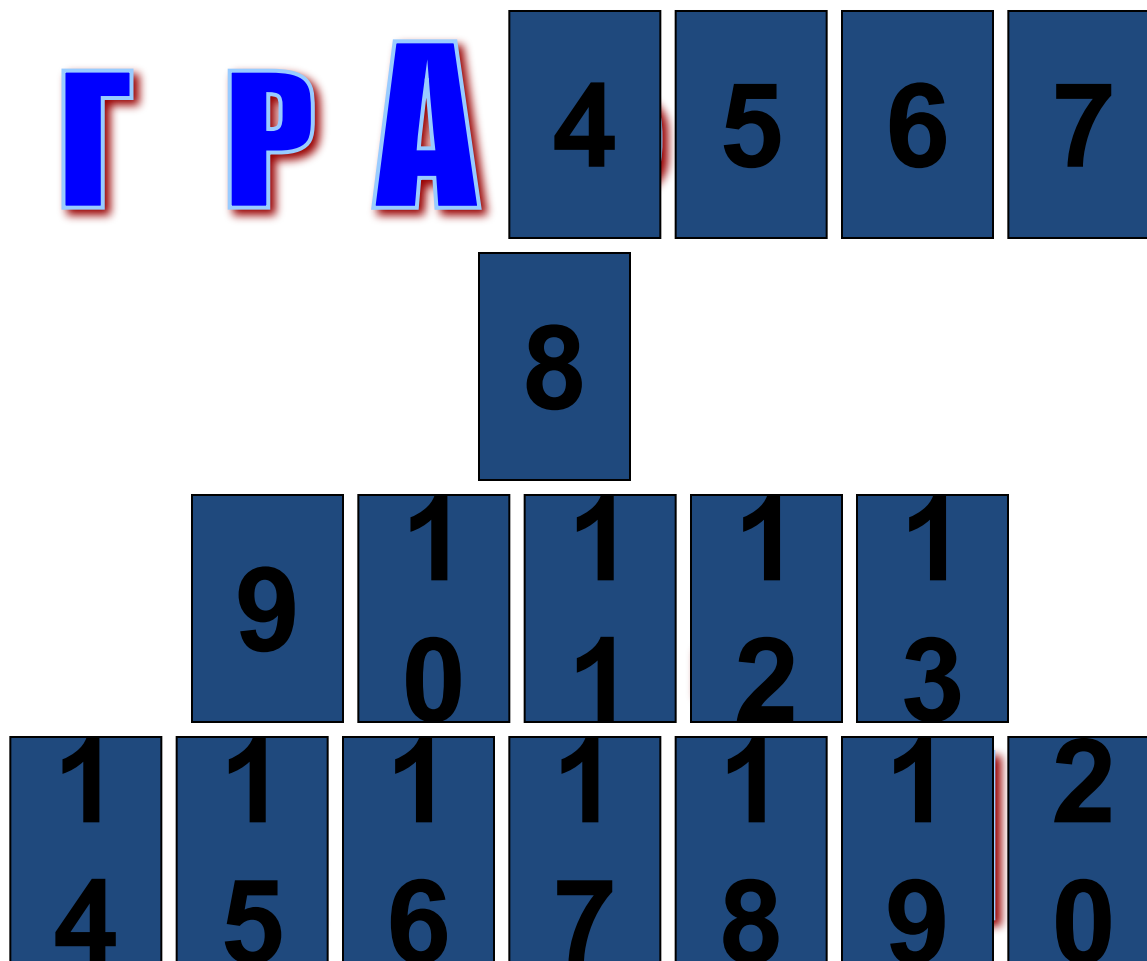
# Тема урока:



# Тема урока:



# Тема урока:



# Тема урока:



# Тема урока:

**Г Р А Ф И** 6 7

8

9

1

1

1

1

0

1

2

3

1

1

1

1

1

1

2

4

5

6

7

8

9

0

# Тема урока:

**Г Р А Ф И К** 7

8

9

1

1

1

1

0

1

2

3

1

1

1

1

1

1

2

4

5

6

7

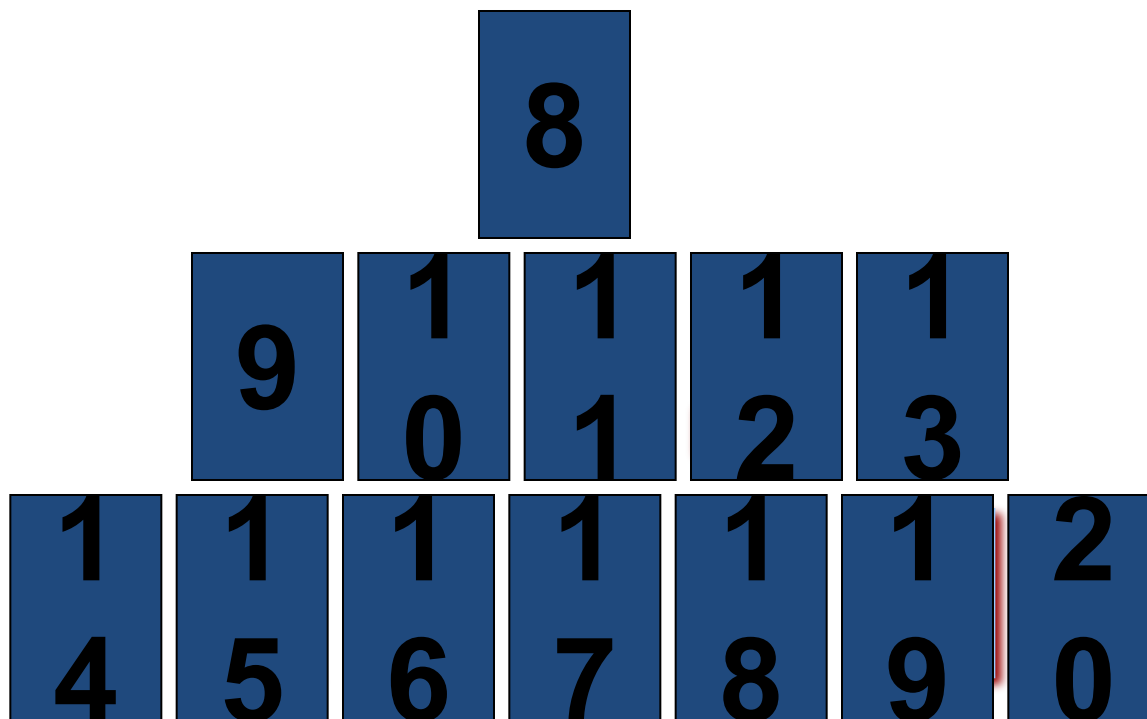
8

9

0

Тема урока:

# Г Р А Ф И К А

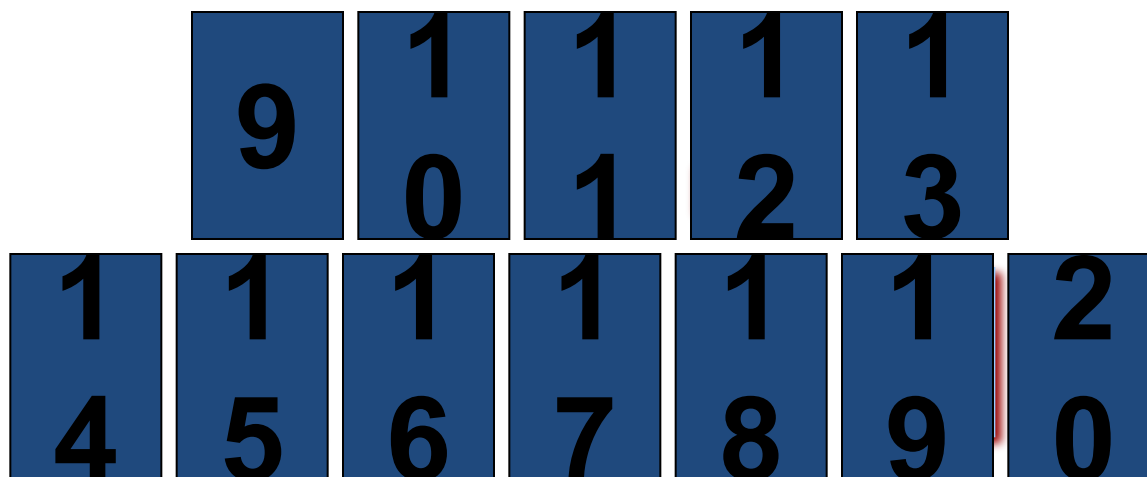




Тема урока:

**Г Р А Ф И К А**

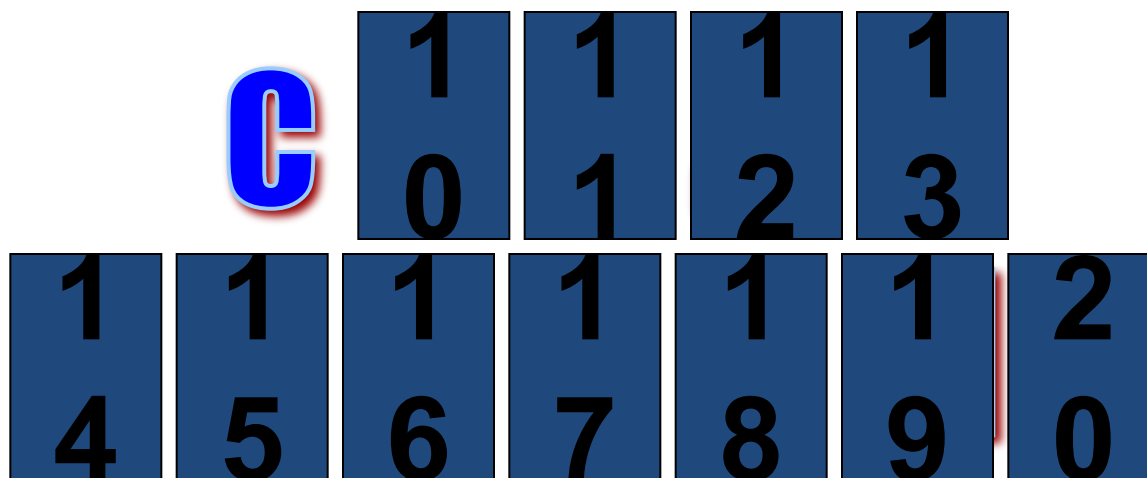
**В**



# Тема урока:

**Г Р А Ф И К А**

**В**



# Тема урока:

**Г Р А Ф И К А**

**В**

**С Р**

1	1	1
1	2	3

1	1	1	1	1	1	2
4	5	6	7	8	9	0

# Тема урока:

**Г Р А Ф И К А**

**В**

**С Р Е**

1	1
2	3

1	1	1	1	1	1	2
4	5	6	7	8	9	0

# Тема урока:

**Г Р А Ф И К А**

**В**

**С Р Е Д**

**1**

**3**

**1**

**1**

**1**

**1**

**1**

**1**

**2**

**4**

**5**

**6**

**7**

**8**

**9**

**0**

Тема урока:

**Г Р А Ф И К А**

**В**

**С Р Е Д Е**

1	1	1	1	1	1	2
4	5	6	7	8	9	0

Тема урока:

**Г Р А Ф И К А**

**В**

**С Р Е Д Е**

<b>п</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>2</b>
	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>0</b>

# Тема урока:

**Г Р А Ф И К А**

**В**

**С Р Е Д Е**

<b>П</b>	<b>А</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>2</b>
		<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>0</b>



# Тема урока:

**Г Р А Ф И К А**

**В**

**С Р Е Д Е**

**П А С**

<b>1</b>	<b>1</b>	<b>1</b>	<b>2</b>
<b>7</b>	<b>8</b>	<b>9</b>	<b>0</b>

# Тема урока:

**Г Р А Ф И К А**

**В**

**С Р Е Д Е**

**П А С К**

<b>1</b>	<b>1</b>	<b>2</b>
<b>8</b>	<b>9</b>	<b>0</b>

# Тема урока:

**Г Р А Ф И К А**

**В**

**С Р Е Д Е**

**П А С К А**

**1**

**2**

**9**

**0**

Тема урока:

**Г Р А Ф И К А**

**В**

**С Р Е Д Е**

**П А С К А Л**

**2  
0**

\*

Тема занятия:

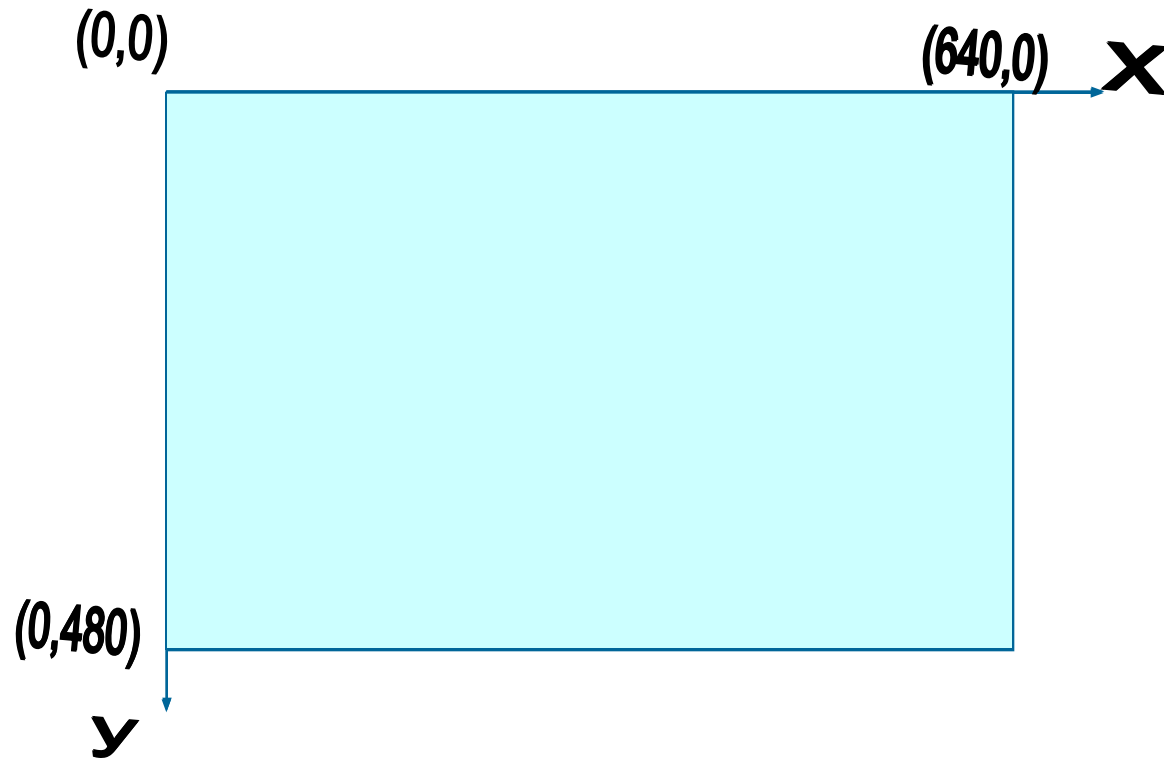
**Г Р А Ф И К А**

**В**

**С Р Е Д Е**

**П А С К А Л Ь**

# Графический режим



Размер экрана 640x480

# Инициализация графического режима

```
Program pr;  
Uses crt, graph;    {использование графического режима}  
Var driver,         {графический драйвер}  
mode:integer;      {графический режим}  
Begin  
Driver:=detect;     {автораспознавание драйвера}  
Initgraph(driver, mode, 'путь к файлу egavga.bgi');  
{операторы и процедуры графических построений}  
Readkey;  
Closegraph;  
End.
```

# Номера цветов

Константа	Значение	Константа	Значение
Black	0 (черный)	DarkGray	8 (темно-серый)
Blue	1 (синий)	LightBlue	9 (светло-синий)
Green	2 (зеленый)	LightGreen	10 (светло-зеленый)
Cyan	3 (голубой)	LightCyan	11 (светло-голубой)
Red	4 (красный)	LightRed	12 (розовый)
Magenta	5 (малиновый)	LightMagenta	13 (светло-малиновый)
Brown	6 (коричневый)	Yellow	14 (желтый)
LightGray	7 (светло-серый)	White	15 (белый)
Blink 128 (мерцание)			



# Установка цвета

**SetColor(Color:integer);**

*Устанавливает основной цвет рисования*

**SetBkColor(Color:integer);**

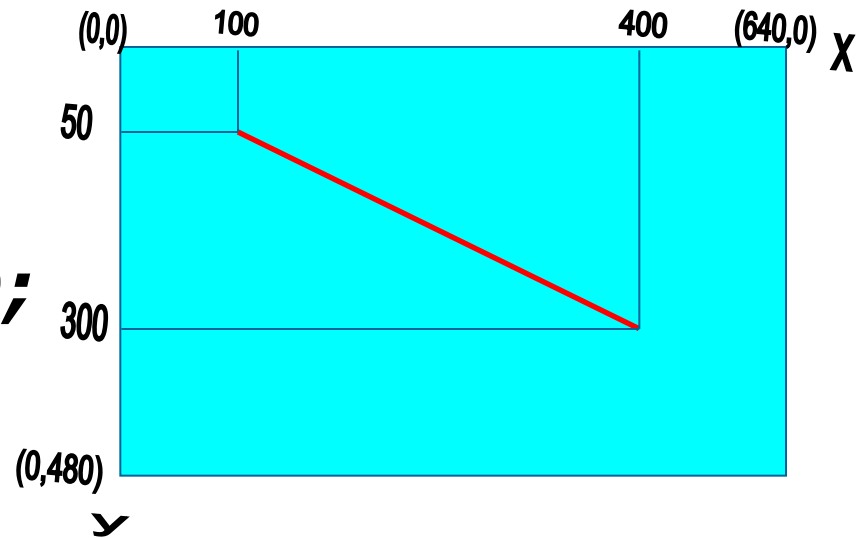
*Устанавливает текущий фоновый цвет*

*Например: нарисовать на экране синего цвета отрезок прямой из точки с координатами(100,50) в точку с координатами(400,300) красного цвета:*

**SetBkColor(3);**

**SetColor(4);**

**LINE(100,50,400,300);**



## **MoveTo (x, y);**

Устанавливает перо в координаты (x,y).

## **PutPixel (x, y, Color);**

Рисует точку с координатами (x,y) цветом Color.

## **GetPixel (x, y): word;**

Возвращает цвет точки с координатами (x,y).

## **SetLineStyle (Ln, 0, T);**

Изменяет параметры контуров. **Ln** - стиль линии (0..3):

**T** - толщина линии: 1 = нормальная; 3 = толстая.

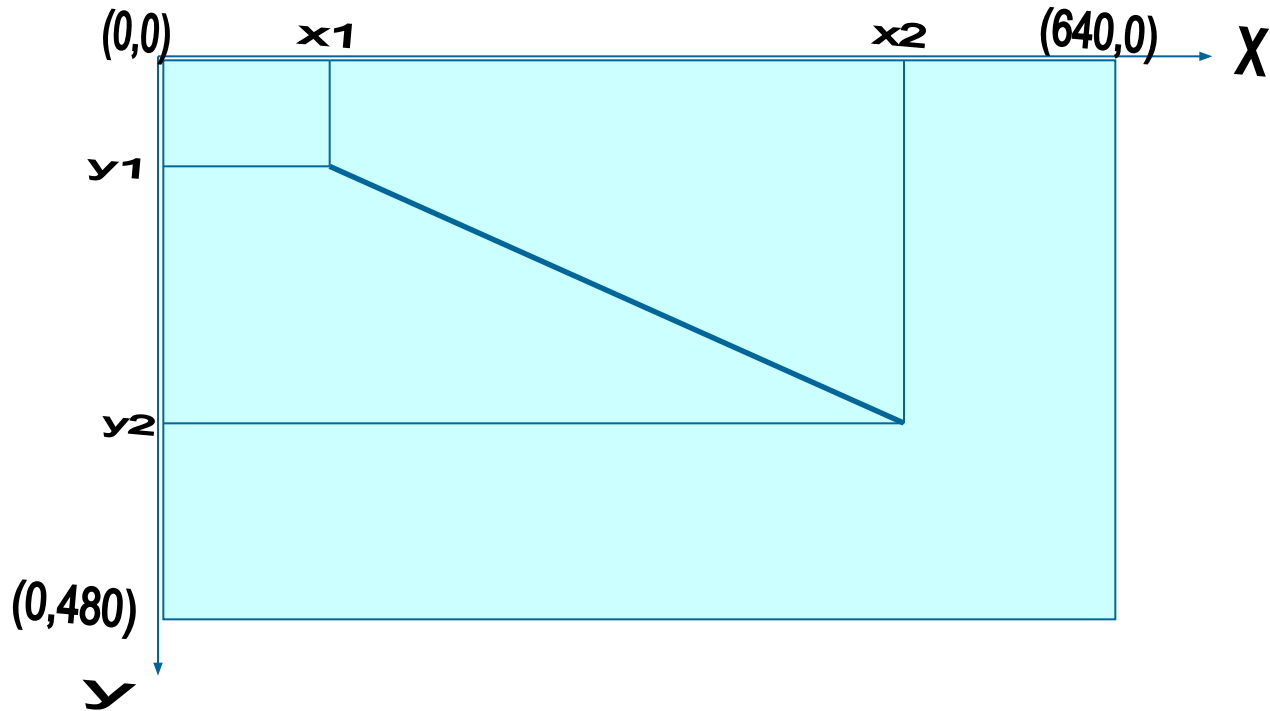
## **SetFillStyle (Type, Color);**

Устанавливает тип (0..11) и цвет штриховки замкнутых фигур

## **ClearDevice;**

Очищает графический экран закрашивая его в текущий цвет фона

# Линия



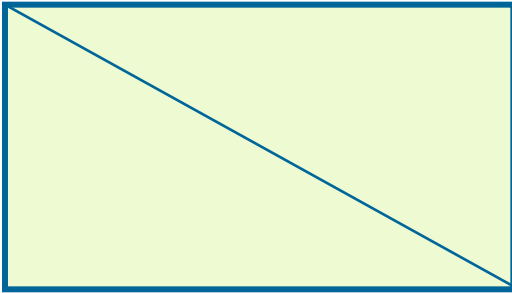
Оператор **LINE(X1,Y1,X2,Y2)**; рисует на экране отрезок прямой из точки с координатами  $(x1,y1)$  до точки с координатами  $(x2,y2)$ .

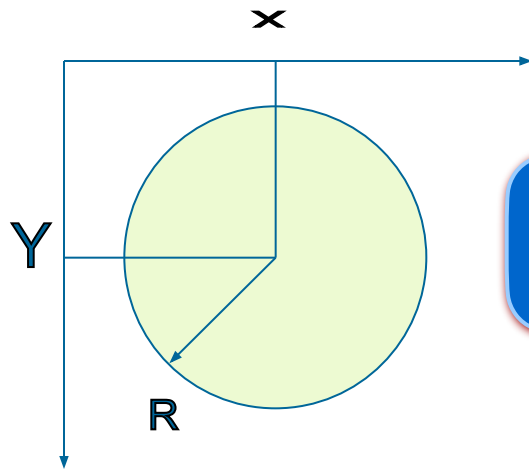
# Прямоугольник

Процедура

**Var(x1,y1,x2,y2)**

выводит на экран закрашенный  
прямоугольник с координатами  
левого верхнего угла (x1,y1) и  
правого нижнего угла (x2,y2).





# ОКРУЖНОСТЬ

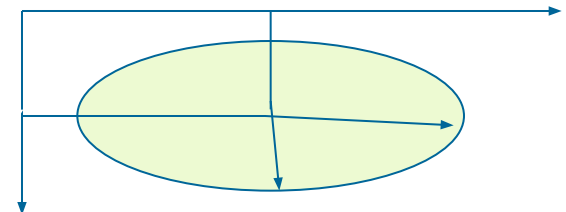
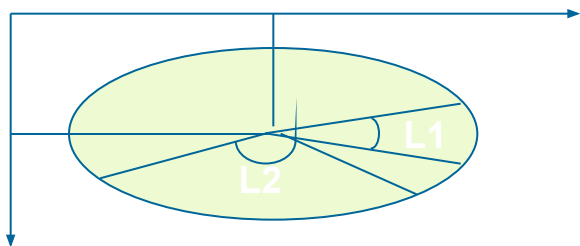
Процедура **CIRCLE(x,y,r)** выводит на экран окружность с центром **(x, y)** и радиусом **r**.

# ЭЛЛИПС

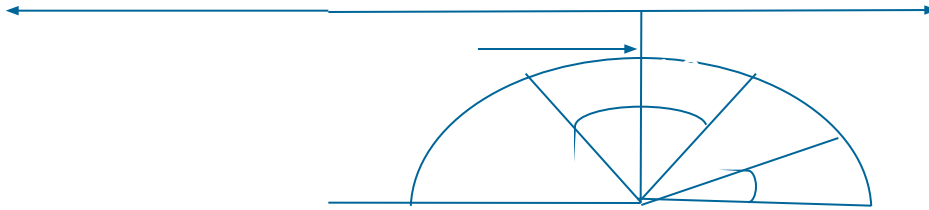
Процедура **ELLIPSE(x,y,L1,L2,Rx,Ry)** рисует эллиптическую дугу от начального угла **L1** до конечного **L2**, используя  $X, Y$  как координаты центра, как горизонтальный (**Rx**) и вертикальный (**Ry**) радиус.

**FillEllipse (x, y, Xr, Yr);**

Рисует закрашенный эллипс с центром в  $(x, y)$  и радиусами  $Xr$  и  $Yr$ .



# Дуга



Процедура **Arc(x,y,L1,L2,r)**  
рисует дугу с центром в точке (x,  
y) от начального угла L1 до  
конечного L2 радиусом r.

Для обеспечения задержки выполнения действия используется процедура:

**DELAY(n);**

где  $n$  – это величина (в миллисекундах) необходимой задержки.

Процедура остановки цикла после нажатия любой клавиши:

**KEYPRESSED;**



# Программа движения окружности:

Program okr;

Uses graph, crt;

Var x, y, r: integer; { параметры окружности }  
dx: integer; { шаг перемещения окружности }  
grdriver, grmode: integer; { графический режим }

Begin

Grdriver:=detect;

Initgraph ( grdriver, grmode, ' c:/BP/BGI' );

X:=0;

Y:=100;

R:=20;

Dx:=2;

Repeat

Setcolor(15);

Circle(x,y,r);

Delay(10000);

Setcolor(0);

Circle(x,y,r);

X:=x+dx;

Until keypressed;

Readln;

Closegraph;

End.



# Домашнее задание:

- 1) Повторение теоретического материала.
- 2) Как Выполнить мерцающую разными цветами окружность? Подумать и записать дополнение к программе.

Спасибо  
за урок!

