

# Программирование на языке Паскаль

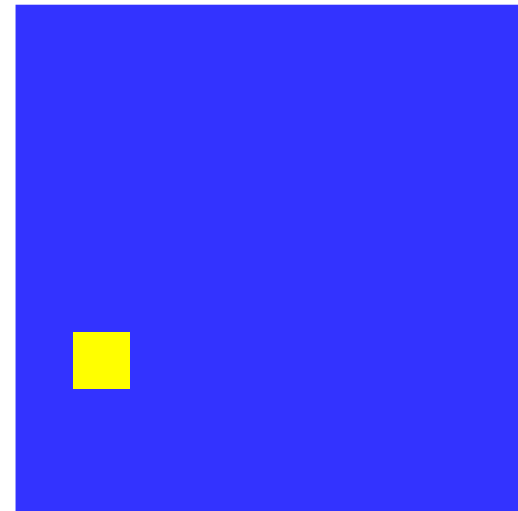
Анимация

# Анимация

---

**Анимация** (англ. *animation*) – оживление изображения на экране.

**Задача:** внутри синего квадрата 400 на 400 пикселей слева направо движется желтый квадрат 20 на 20 пикселей. Программа останавливается, если нажата клавиша *Esc* или квадрат дошел до границы синей области.



**Проблема:** как изобразить перемещение объекта на экране?

**Привязка:** состояние объекта задается координатами  $(x,y)$

**Принцип анимации:**

1. рисуем объект в точке  $(x,y)$
2. задержка на несколько миллисекунд
3. стираем объект
4. изменяем координаты  $(x,y)$
5. переходим к шагу 1

# Как "поймать" нажатие клавиши?

---

**Событие** – это изменение в состоянии какого-либо объекта или действие пользователя (нажатие на клавишу, щелчок мышкой).

**IsEvent** – логическая функция, которая определяет, было ли какое-то действие пользователя.

**Event** – процедура, которая определяет, какое именно событие случилось.

```
if IsEvent then begin
  Event(k, x, y);
  if k = 1 then
    writeln('Клавиша с кодом ', x)
  else { k = 2 }
    writeln('Мышь: x=', x, ' y=', y);
end;
```

var k, x, y: integer;

# Как выйти из цикла при нажатии *Esc*?

```
program qq;  
var stop: boolean;  
    k, code, i: integer;  
begin  
    stop := False;  
    repeat  
        if IsEvent then begin  
            Event(k, code, i);  
            if (k = 1) and (code = 27) then  
                stop := True;  
        end;  
        ...  
    until stop;  
end.
```

True, если надо  
остановиться

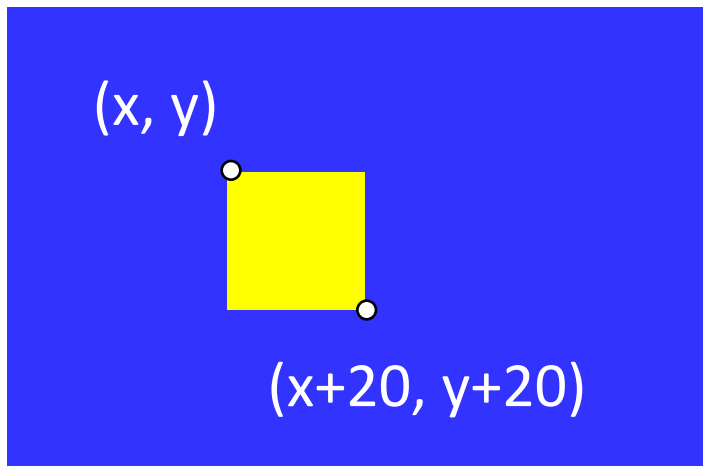
запуск цикла

если что-то  
произошло...

что произошло?

если нажата клавиша с  
кодом 27 (*Esc*), то стоп

# Процедура (рисование и стирание)



## Идеи

- одна процедура рисует и стирает
- стереть = нарисовать цветом фона
- границу квадрата отключить (в основной программе)

рисовать (True) или нет (False)?

```
procedure Draw(x, y: integer; flag: boolean);
begin
  if flag then
    Brush(1, 255, 255, 0)
  else
    Brush(1, 0, 0, 255);
  Rectangle(x, y, x+20, y+20);
end;
```

рисуем: цвет кисти – желтый

стираем: цвет кисти – синий

только заливка!

# Полная программа

```
program qq;  
var x, y, k, code, i: integer;  
    stop: boolean;  
procedure Draw(x,y: integer; flag: Boolean);  
begin  
    ...  
end;  
begin  
    Brush(1, 0, 0, 255);  
    Rectangle(10, 10, 400, 400);  
    Pen(0, 0, 0, 255);  
    x := 10; y := 200; stop := false;  
    repeat  
        if IsEvent then begin  
            ...  
        end;  
        Draw(x, y, True);  
        Delay(10);  
        Draw(x, y, False);  
        x := x + 1;  
        if x >= 400-20 then stop := true;  
    until stop;  
end.
```

процедура

синий фон

отключить границу

начальные условия

ВЫХОД ПО  
клавише *Esc*

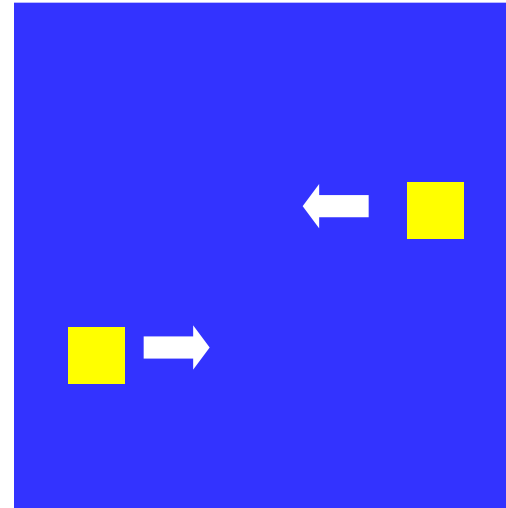
ждем 10 мс

ВЫХОД ПРИ  
касании границы

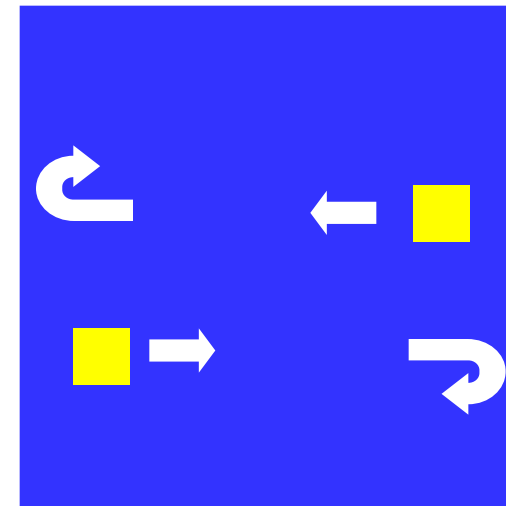
# Задания

---

«4»: Два квадрата двигаются в противоположном направлении:



«5»: Два квадрата двигаются в противоположном направлении и отталкиваются от стенок синего квадрата:



# Управление клавишами

**Задача:** жёлтый квадрат внутри синего квадрата управляется клавишами-стрелками. Коды клавиш:

влево – 37      вверх – 38      Esc – 27  
вправо – 39      вниз – 40

**Проблема:** как изменять направление движения?

**Решение:**

```
if IsEvent then begin
  Event ( k, code, i);
  if k = 1 then begin
    case code of
      37: x := x - 1; 38: y := y - 1;
      39: x := x + 1; 40: y := y + 1;
      27: stop := True;
    end;
  end;
end;
```

если было нажатие на клавишу, ...



# Программа

```
program qq;  
var x, y, k, code, i: integer;  
    stop: boolean;
```

процедура

```
procedure Draw(x, y: integer; flag: Boolean);  
begin  
    ...  
end;
```

```
begin
```

```
    ...
```

ОСНОВНОЙ ЦИКЛ

```
repeat  
    Draw(x, y, True);  
    Delay(20);  
    Draw(x, y, False);
```

```
    if IsEvent then begin  
        ...  
    end;
```

обработка  
событий

```
until stop;
```

```
end.
```



Что плохо?

# Как убрать мигание?

---

**Проблема:** даже если не нажата никакая клавиша, квадрат перерисовывается через каждые 20 мс (мигание!)

**Что хочется:** не перерисовать квадрат, если не было никакого события

**Решение:** нарисовать квадрат и ждать события

**Новая проблема:** как ждать события?

**Решение новой проблемы:** пустой цикл "пока не случилось событие, ничего не делай":

```
while not IsEvent do;
```

# Программа

```
program qq;  
var x, y, k, code, i: integer;  
    stop: boolean;
```

процедура

```
procedure Draw(x, y: integer; flag: Boolean);  
begin  
    ...  
end;
```

```
begin
```

```
    ...
```

рисуем квадрат

```
repeat
```

```
    Draw(x, y, True);
```

```
    while not IsEvent
```

```
do;
```

```
    Draw(x, y, False);
```

```
    Event(k, code, i);
```

```
    ...
```

```
until stop;
```

ждем события

только теперь стираем

```
end.
```

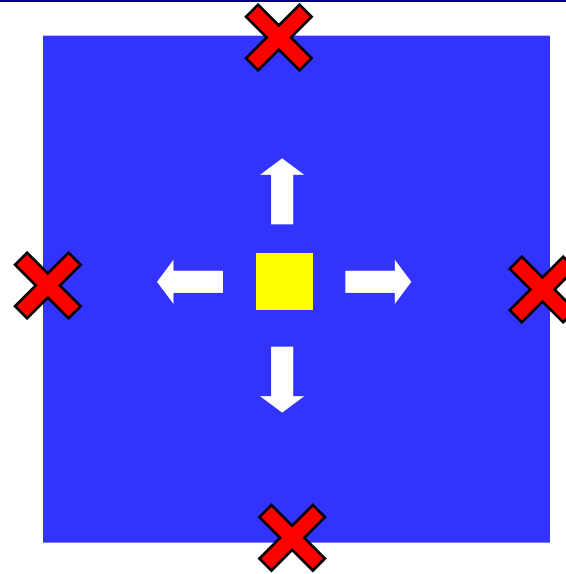


Что можно улучшить?

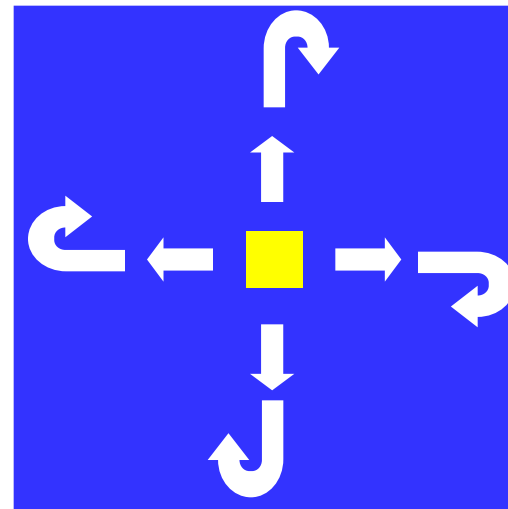
# Задания

---

«4»: Квадрат двигается при нажатии стрелок, однако не может выйти за границы синего квадрата:



«5»: Квадрат непрерывно двигается, при нажатии стрелок меняет направление и отталкивается от стенок синего квадрата:



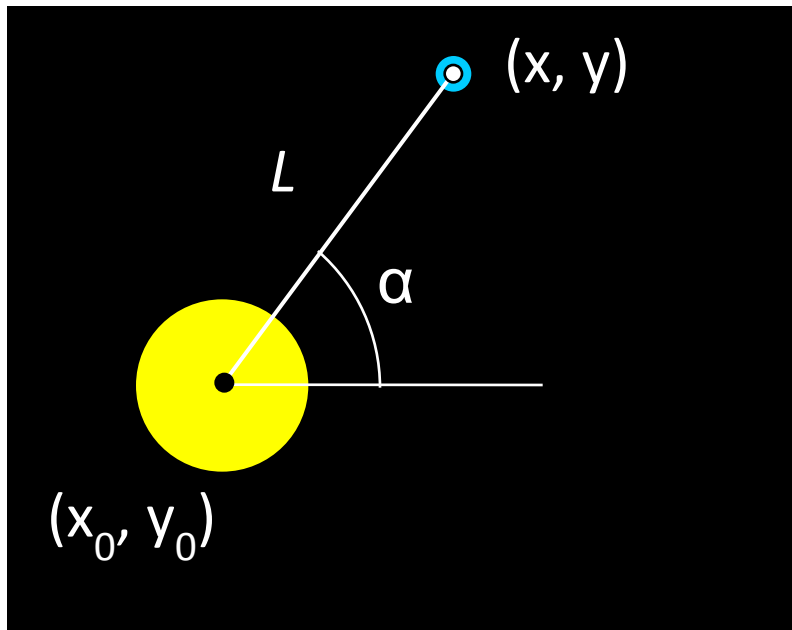
# Вращение

---

**Задача:** изобразить модель вращения Земли вокруг Солнца.

**Проблема:** движение по окружности, как изменять координаты?

**Решение:** использовать в качестве независимой переменной (менять в цикле) угол поворота  $\alpha$



$$x = x_0 + L \cdot \cos(\alpha)$$

$$y = y_0 - L \cdot \sin(\alpha)$$

# Процедура

рисовать (True) или нет (False)?

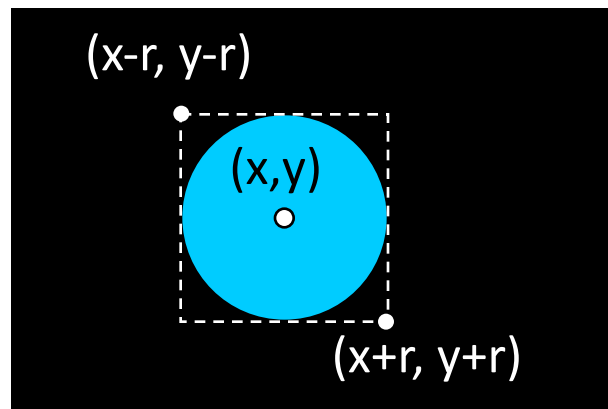
```
procedure Draw(x, y: integer; flag: boolean);  
const r = 10;  
begin  
  if flag then  
    Brush(1, 100, 100, 255)  
  else  
    Brush(1, 0, 0, 0);  
  Ellipse(x-r, y-r, x+r, y+r);  
end;
```

радиус Земли

рисуем: цвет кисти – голубой

стираем: цвет кисти – черный

только заливка!



# Константы и переменные

```
program qq;
const rSun = 60;      { радиус Солнца}
      L  = 150;      { радиус орбиты Земли }
      x0 = 200;      { координаты центра Солнца}
      y0 = 200;
var x, y,             { координаты Земли }
    k, code, i: integer; { для Event }
    a, ha: real;      { угол поворота, шаг }
stop: boolean; { признак остановки программы }
procedure Draw(x, y: integer; flag:
Boolean);
begin
    ...
end;
begin
    ...
end.
```

# Основная программа

```
program qq;
...
begin
  Brush(1, 0, 0, 0);  Fill(1,1);
  Brush(1, 255, 255, 0);
  Ellipse(x0-rSun, y0-rSun, x0+rSun, y0+rSun);
  a := 0; ha := 1*pi/180; { начальный угол, шаг 1° за 100 мс }
  stop := false;
  Pen(0,0,0,0);      { отключаем контуры }
  repeat
    x := round(x0 + L*cos(a));
    y := round(y0 - L*sin(a));
    Draw(x, y, True);
    Delay(100);
    Draw(x, y, False);
    if IsEvent then begin
      Event(k, code, i);
      if (k = 1) and (code = 27) then stop := true;
    end;
    a := a + ha;
  until stop;
end.
```

залить фон черным

рисует Солнце

новые координаты

ждем 100 мс

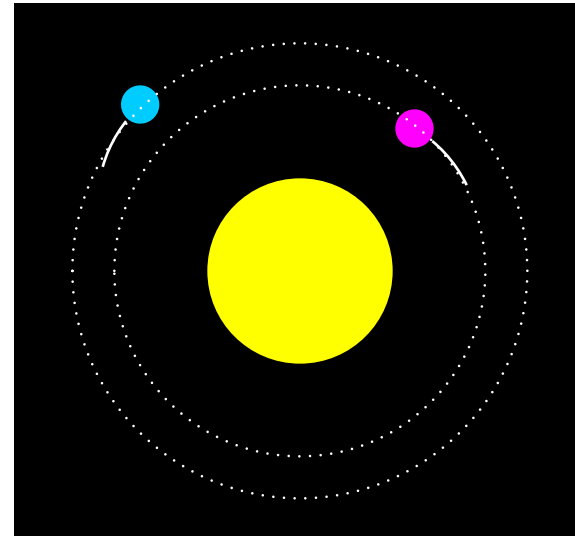
поворот на ha



# Задания

---

«4»: Изобразить модель Солнца с двумя планетами, которые вращаются в противоположные стороны:



«5»: Изобразить модель системы Солнце-Земля-Луна:

