

# Линейные программы на Паскале



# Основные понятия:

---

- Программирование- раздел информатики, посвященный методам разработки программ управления компьютером.
- Программист-человек, создающий программы.
- Программа –алгоритм решения задачи, записанный на языке программирования.

# Этапы решения вычислительных задач:

---

- Постановка задачи
- Математическая формализация
- Построение алгоритма
- Составление программы на языке программирования
- Отладка и тестирование программы
- Проведение расчетов и анализ результатов

# Структура программы.

## Линейные алгоритмы.

---

### **Заголовок программы**

```
program <имя программы>;
```

### **Раздел описания переменных**

```
Var <список переменных> : <тип>;
```

### **Раздел операторов**

```
Begin
```

```
    <список операторов>;
```

```
End.
```



# Операторы:

---

1. Присваивания `x:=5`

2. Ввода

`Read( x,y)`      `readln( x,y)`

3. Вывода

`Write ('Z=',z)`      `writeln ('результат равен',  
z)`

# Разветвляющийся алгоритм.

---

Задание 1. По заданным предложениям угадать известную пословицу.

- Если два дела делать, то ничего не получится
- Если делать дело не торопясь, то быстрее его закончишь
- Если не будешь трудиться, то не сварешь ухи

---

За двумя зайцами погонишься – ни одного не поймаешь.

Тише едешь – дальше будешь.

Без труда не вынешь рыбку из пруда.



## □ **Если – то**

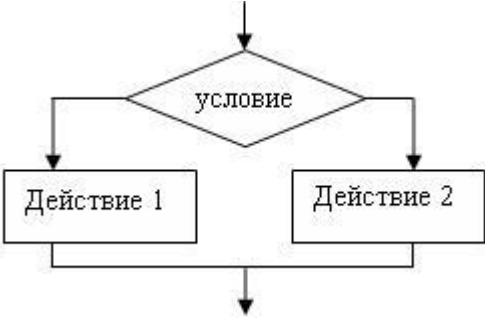
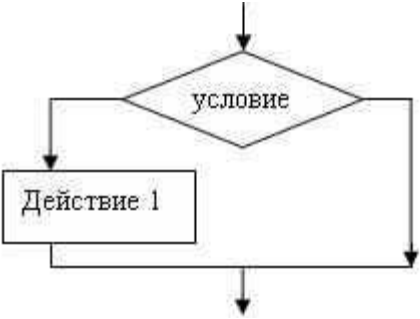
Эта конструкция описывает разветвляющийся алгоритм.

- Примеры (если плохая погода, то беру зонт; если урок физкультуры, то одеваю спортивный костюм и т.д.)

- 
- *Ветвление – это такая организация алгоритма, при которой в зависимости от выполнения или невыполнения условия совершается либо одно, либо другое действие.*

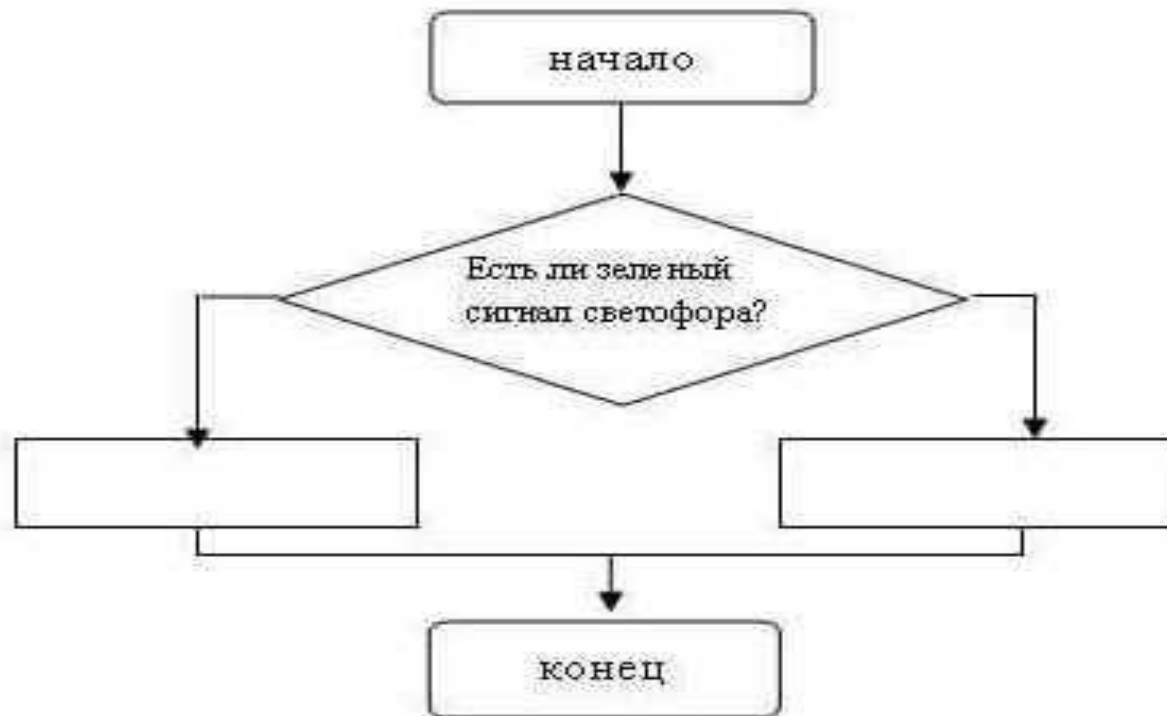


# Ветвление

| Полное   | Неполное   |
|--|--|
| Словесное описание ветвления   |  |
| Если <условие> то <действие 1><br>иначе <действие 2>;  | Если <условие> то <действие 1>;  |
| Блок -схема  |  |
| Описание в языке Паскаль   |  |
| if<условие> then <действие 1> else<br><действие 2>;  | if <условие> then <действие 1>;  |
|  <pre>graph TD; Start(( )) --&gt; Condition{условие}; Condition --&gt; Action1[Действие 1]; Condition --&gt; Action2[Действие 2]; Action1 --&gt; Exit(( )); Action2 --&gt; Exit;</pre> |  <pre>graph TD; Start(( )) --&gt; Condition{условие}; Condition --&gt; Action1[Действие 1]; Condition --&gt; Exit(( )); Action1 --&gt; Exit;</pre> |

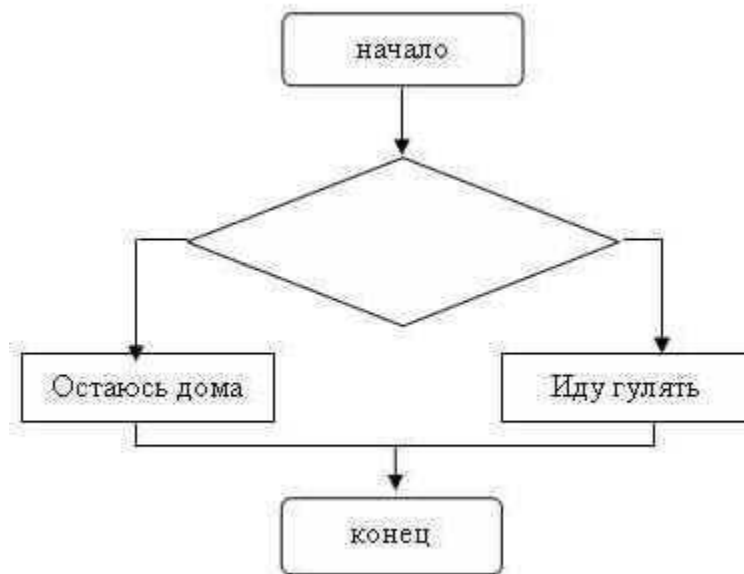
# Расставь в алгоритме действия

---



Придумай и напиши вопрос, который может отменить прогулку.

---



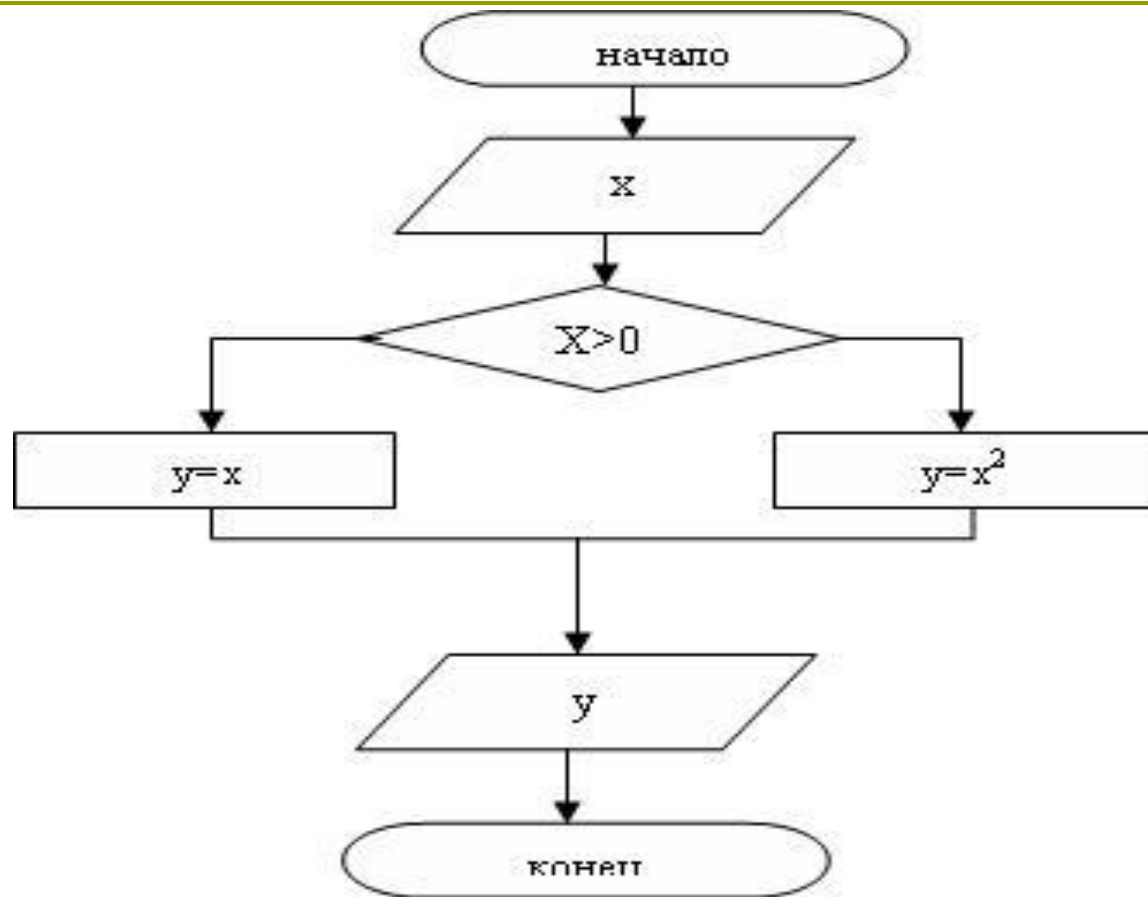
- 
- Вычислить значение функции заданной следующим образом:

$$y = \begin{cases} x, & \text{при } x > 0 \\ x^2, & \text{при } x \leq 0 \end{cases}$$

- Исходные данные:  $x$
- Результат:  $y$
- Составляем блок-схему:

# Блок-схема

---



# Программа

---

```
Program funkciy;  
  Var x,y: integer;  
Begin  
  Writeln ('введите значение x');  
  Readln (x);  
  If x>0 then y:=x  
  else y := x*x;  
  writeln('значение функции ', y);  
  readln;  
end.
```

- 
- Составить программу определяющую является ли заданное число четным или нечетным.