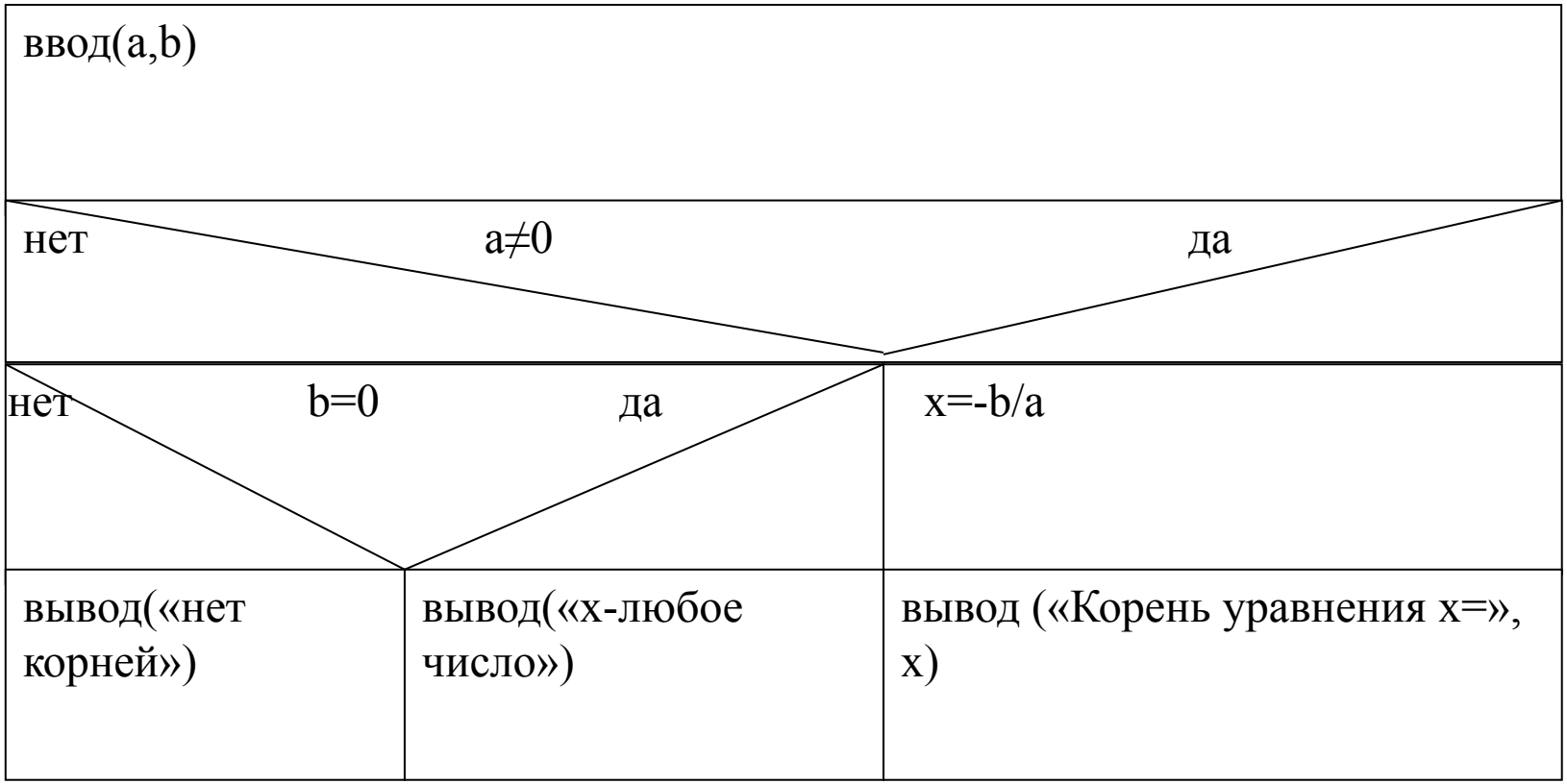
Алгоритм (диаграмма Нейсси-Шнейдермана)



Сравнение различных способов записи алгоритмов и программ

1. Наиболее компактны блок-схема и диаграмма Нейсси-Шнейдермана.
2. Блок-схема наглядна для небольших алгоритмов. Для больших задач это преимущество теряется.
3. При использовании блок-схем можно получить не структурный алгоритм.
4. Псевдокод достаточно компактен и нагляден. По псевдокоду удобнее всего кодировать (его конструкции близки к языкам программирования, особенно Pascal).

Свойства алгоритмов

- **определенность** - однозначная определенность результатов выполнения каждого шага алгоритма;
- **конечность** - результаты выполнения каждого шага алгоритма должны достигаться за конечное время;
- **результативность** - получение конечного результата при любых исходных данных;
- **массовость** - возможность использования алгоритма для некоторого класса исходных данных;
- **правильность** - получение правильных результатов решения поставленной задачи. Говорят, что алгоритм содержит ошибки, если можно указать такие исходные данные или условия, при которых выполнение алгоритма либо не завершается вообще, либо не будет получено никаких результатов, либо полученные результаты окажутся неправильными;
- **надежность** – получение результатов при недопустимых исходных данных.

Общий вид алгоритма

алг «название алгоритма»
нач
    <команды>
кон

<перечисление переменных
с указанием их типов>

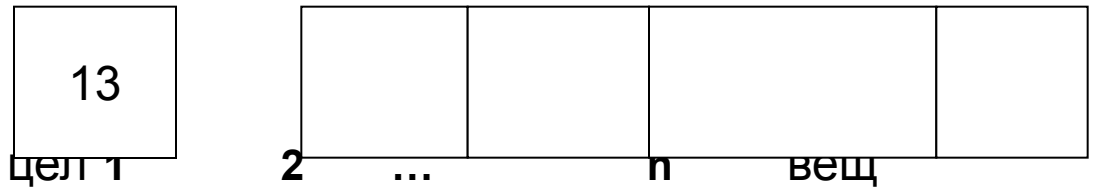
Понятие переменной

<имя >

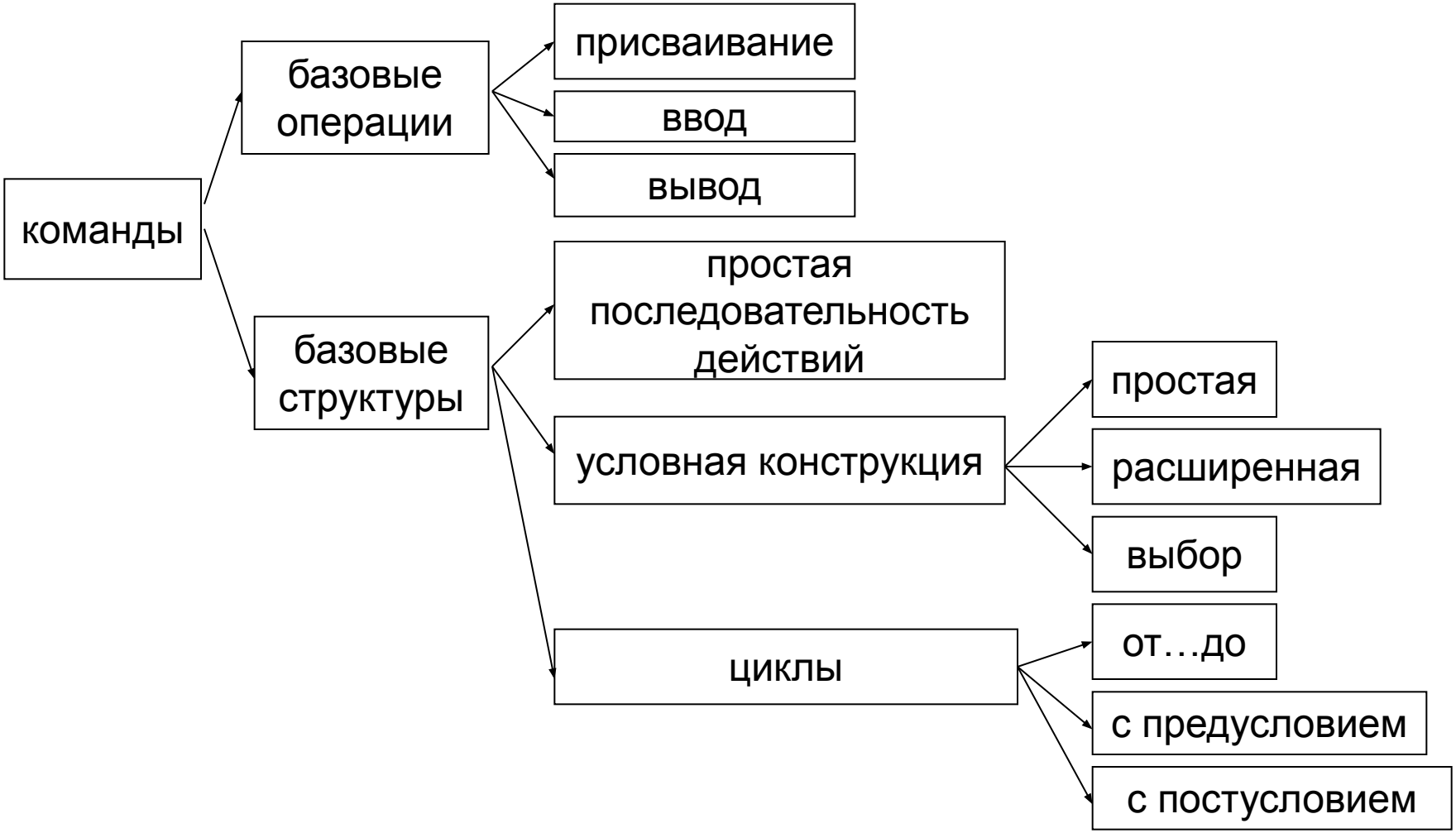
[<значение>]

<тип>

a b



Классификация команд



Операция присваивания

<имя переменной>:=<выражение>

Порядок выполнения

1. Вычисляется значение выражения, расположенного справа от знака :=.
2. Полученное значение присваивается переменной, имя которой расположено слева от знака:=.

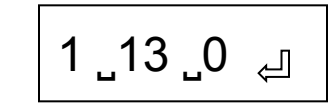
Операция присваивания

Операция ввода

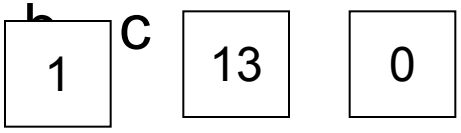
Общий вид:

ввод (<список ввода>)

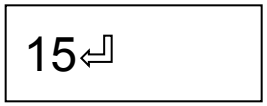
1. ввод (a, b, c)



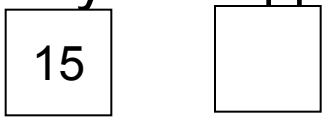
a b c



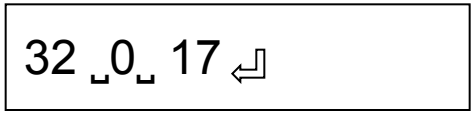
2. ввод (x, y)



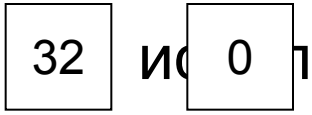
x y ожидается ввод недостающих данных



3. ввод(d, e)



d e лишние данные либо игнорируются, либо



используются при следующем вводе

Операция вывода

Общий вид

вывод (<список вывода>)

1. вывод (“a=“, a, “b=“,b)

```
a=1b=13
```

2. вывод (“a=“, a, “, ␣b=“,b)

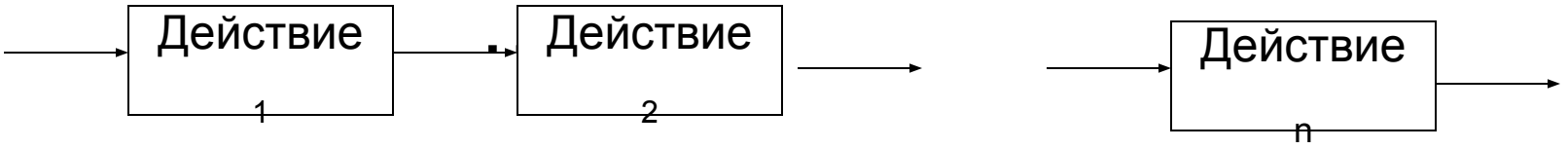
```
a=1, ␣ b=13
```

Форматы в списке вывода не отображаются

Основные алгоритмические конструкции

Простая последовательность действий

Блок-схема



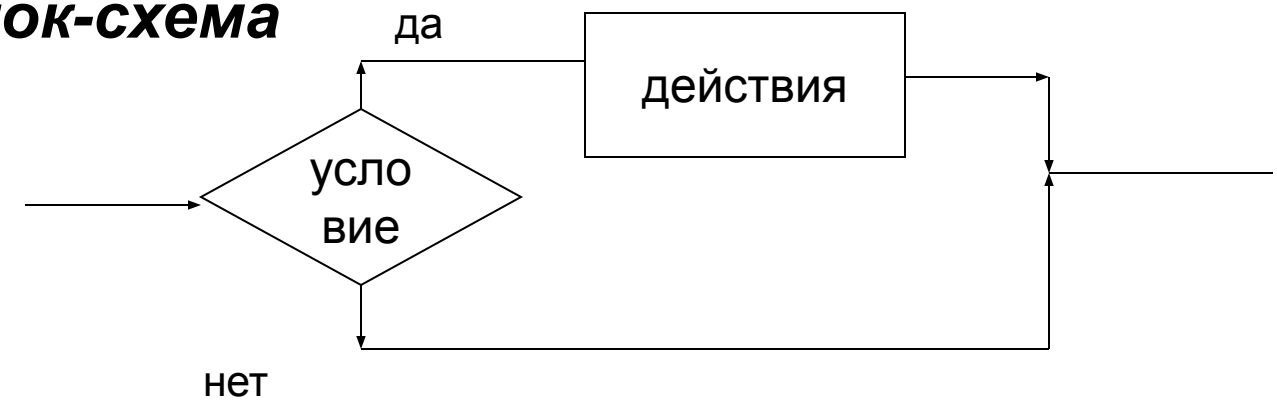
Псевдокод

<действие $_1$>
<действие $_2$>
…
<действие $_n$>

Условная конструкция (выбор)

- Простейшая

Блок-схема



Псевдокод

<u>если</u> <условие> <u>то</u>

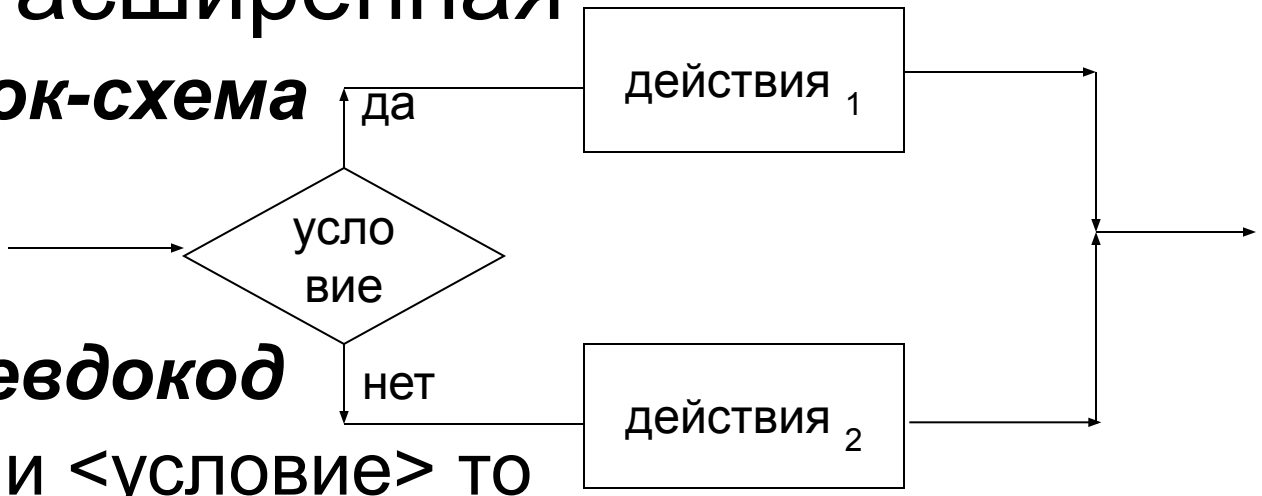
    <действия>

<u>все</u>

Условная конструкция (выбор)

- Расширенная

Блок-схема



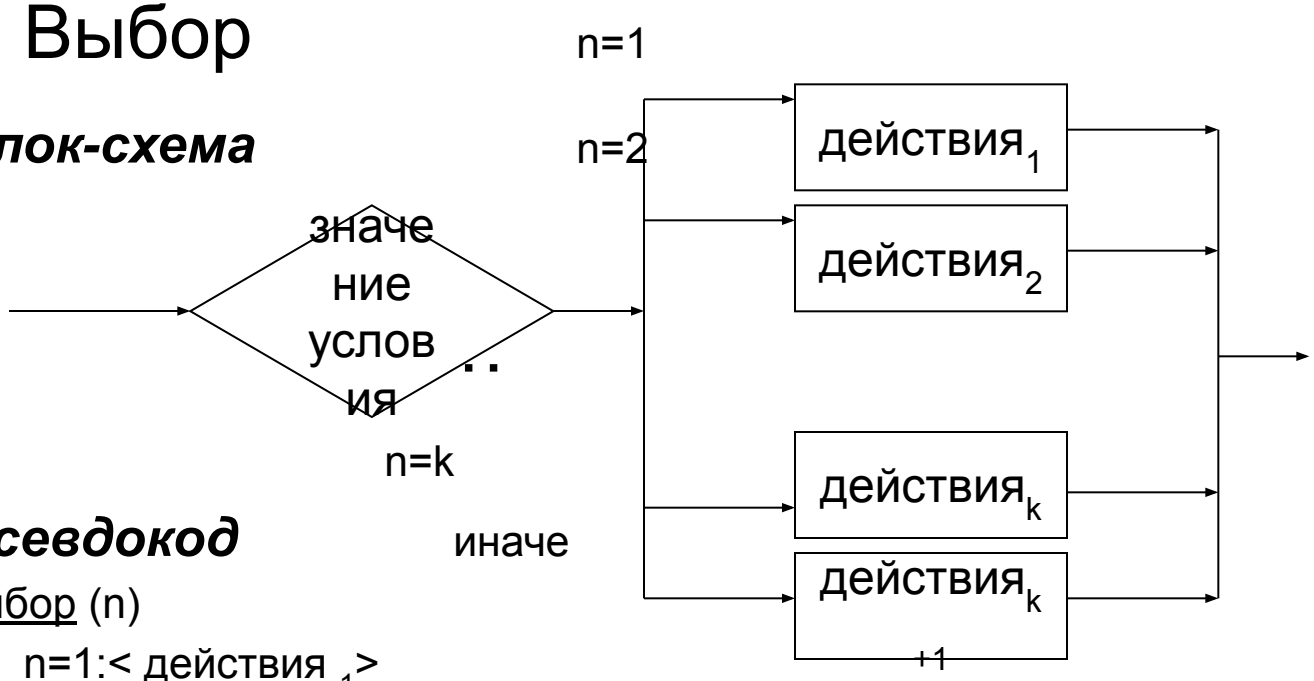
Псевдокод

если <условие> то
    <действия 1>
иначе
    <действия 2>
все

Условная конструкция (выбор)

- Выбор

Блок-схема



Псевдокод

выбор (n)

n=1:< действия $_1$>

n=2:< действия $_2$>

…

n=k:< действия $_k$>

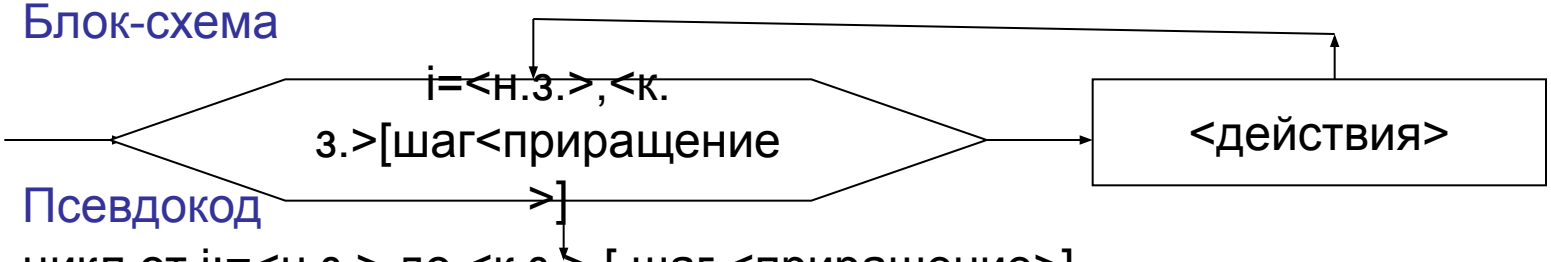
[иначе < действия $_{k+1}$>]

квыбор

Циклы

- От … до (со счетчиком)

Блок-схема



Псевдокод

цикл от i:=<н.з.> до <к.з.> [шаг <приращение>]
    <действия>
кц

, где

i – переменная (счетчик) цикла,

<н.з.> - начальное значение счетчика,

<к.з.> - конечное значение счетчика,

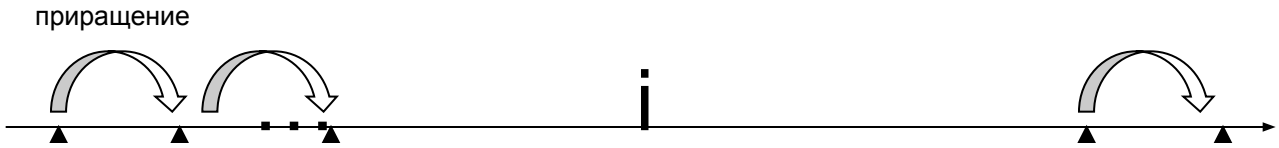
<приращение> - шаг, с которым изменяется значение счетчика.

Если шаг не указан, то он равен 1.

Циклы

- Цикл от… до

при <приращение> >0



<н.з.> <к.з.>

при <приращение> <0

<к.з.> <н.з.>