

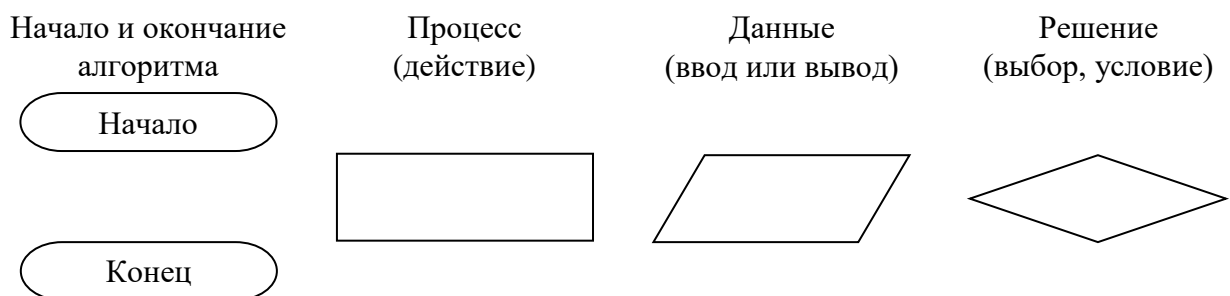
## Изображение алгоритмов при помощи блок-схем

Несколько определений термина «алгоритм»:

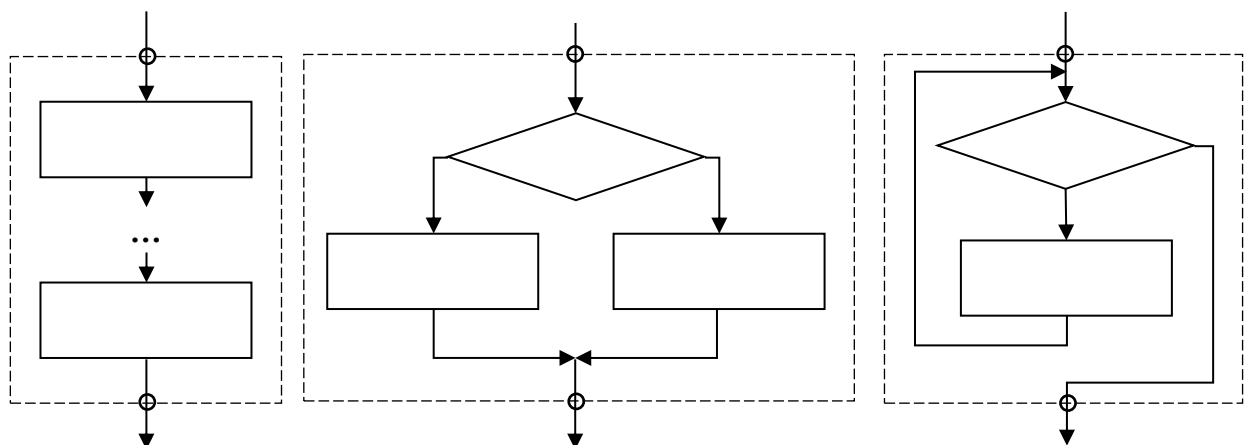
- Алгоритм – это последовательность действий, выполнение которых приводит к заданному результату.
- Алгоритм – это система правил, сформулированная на понятном исполнителю языке, которая определяет процесс перехода от допустимых исходных данных к некоторому результату и обладает свойствами массовости, конечности, определенности, детерминированности.
- Алгоритм – это набор инструкций, описывающих порядок действий исполнителя для достижения некоторого результата.

Понятие алгоритма относится не только к программированию и исполнитель – не обязательно компьютер, например, арифметические алгоритмы умножения и деления «столбиком», кулинарный рецепт как алгоритм приготовления блюда и др. Но широкое применение термин «алгоритм» после математики получил с развитием программирования. Алгоритм может быть представлен различными способами: в виде текстового описания, пошаговой инструкции и др. Текст программы тоже является алгоритмом, записанным на «понятном» для компьютера языке. Наиболее известны так называемые «блок-схемы алгоритмов», которые регламентированы ГОСТ 19.701-90 «Единая система программной документации. Схемы алгоритмов, программ, данных и систем» в разделе «Схемы программ» (этот стандарт идентичен международному стандарту ISO 5807-85).

Основными блоками при изображении алгоритмов являются следующие:



В структурном программировании используется 3 элементарных конструкции: последовательность, выбор (условие, ветвление), повторение (цикл). Каждая из этих конструкция является завершённой и может заменить собой любой процедурный блок (действие) с одним входом и одним выходом, например:

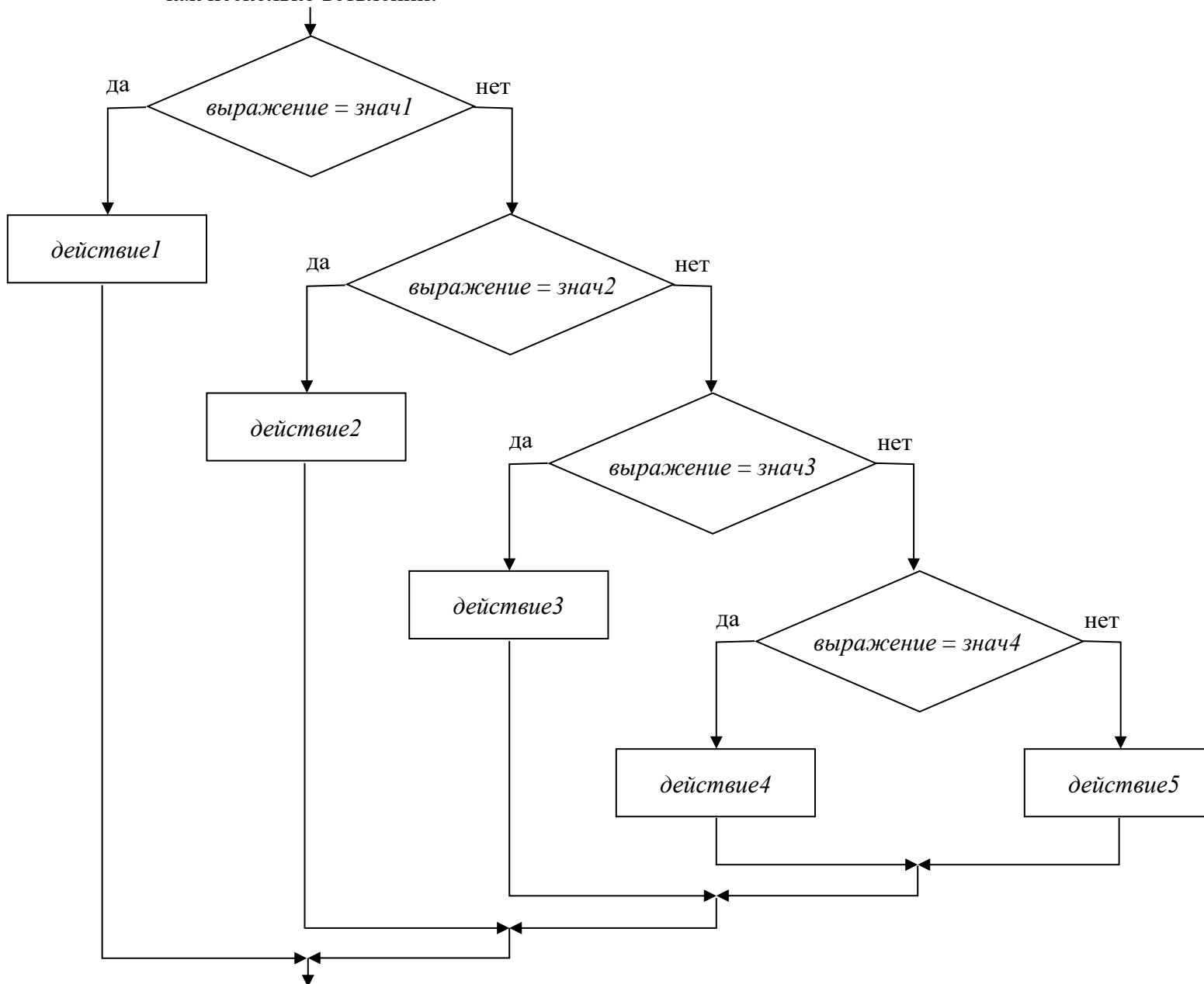


Ниже приведены примеры изображения в виде блок-схем некоторых конструкций языка С.

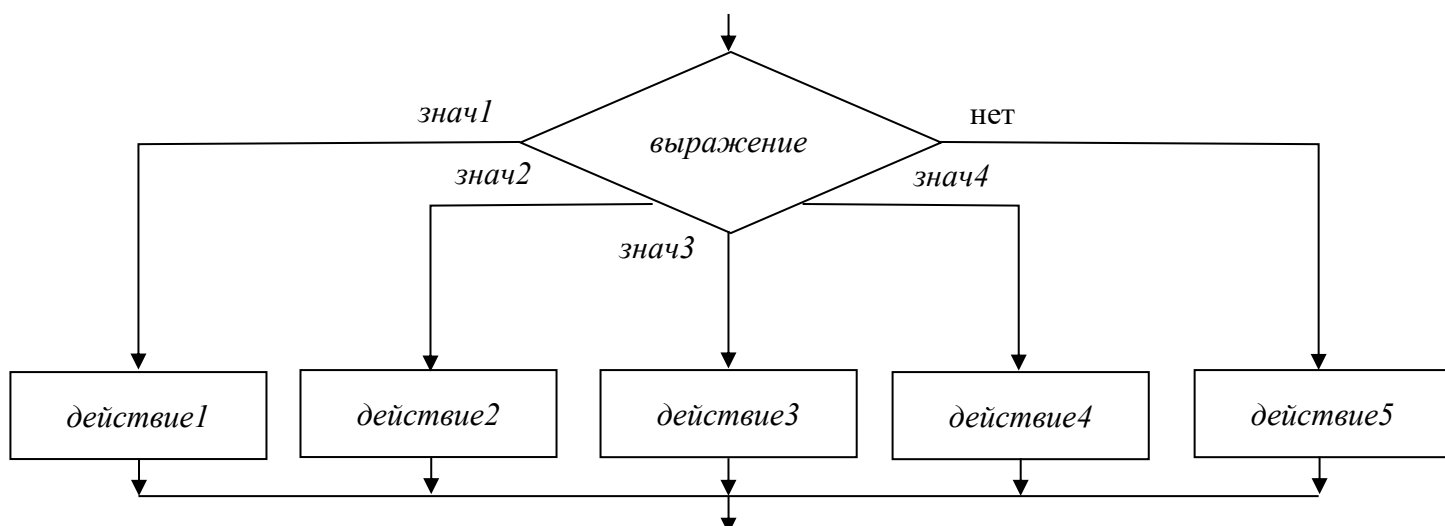
## Изображение оператора switch вида:

```
switch (выражение)
{
    case знач1:
        действие1
        break;
    case знач2:
        действие2
        break;
    case знач3:
        действие3
        break;
    case знач4:
        действие4
        break;
    default:
        действие5
        break;
}
```

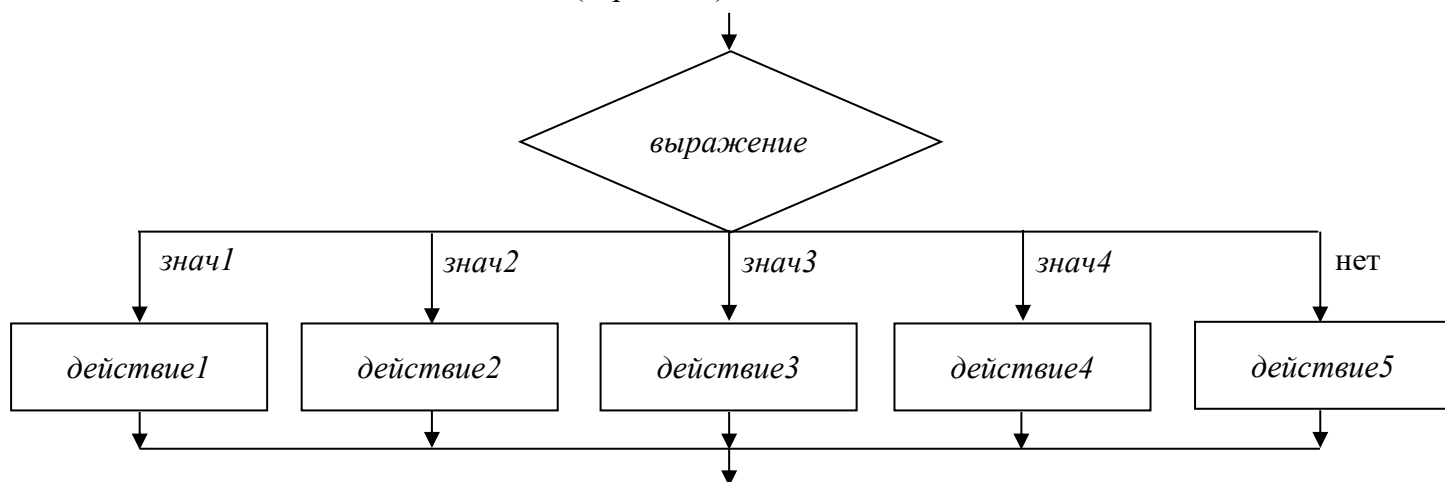
— как несколько ветвлений:



– как множественное ветвление (вариант 1):



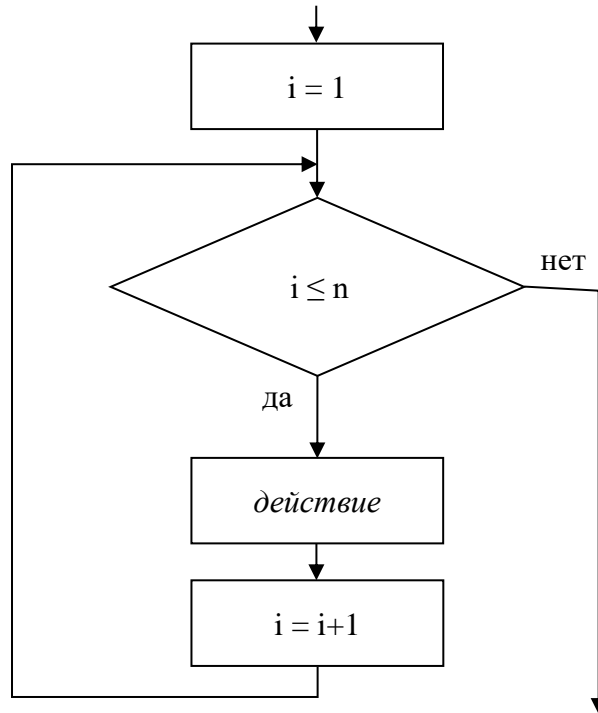
– как множественное ветвление (вариант 2):



## Изображение циклов

```
i = 1;  
while (i <= n)  
{  
    действие  
    i++;  
}
```

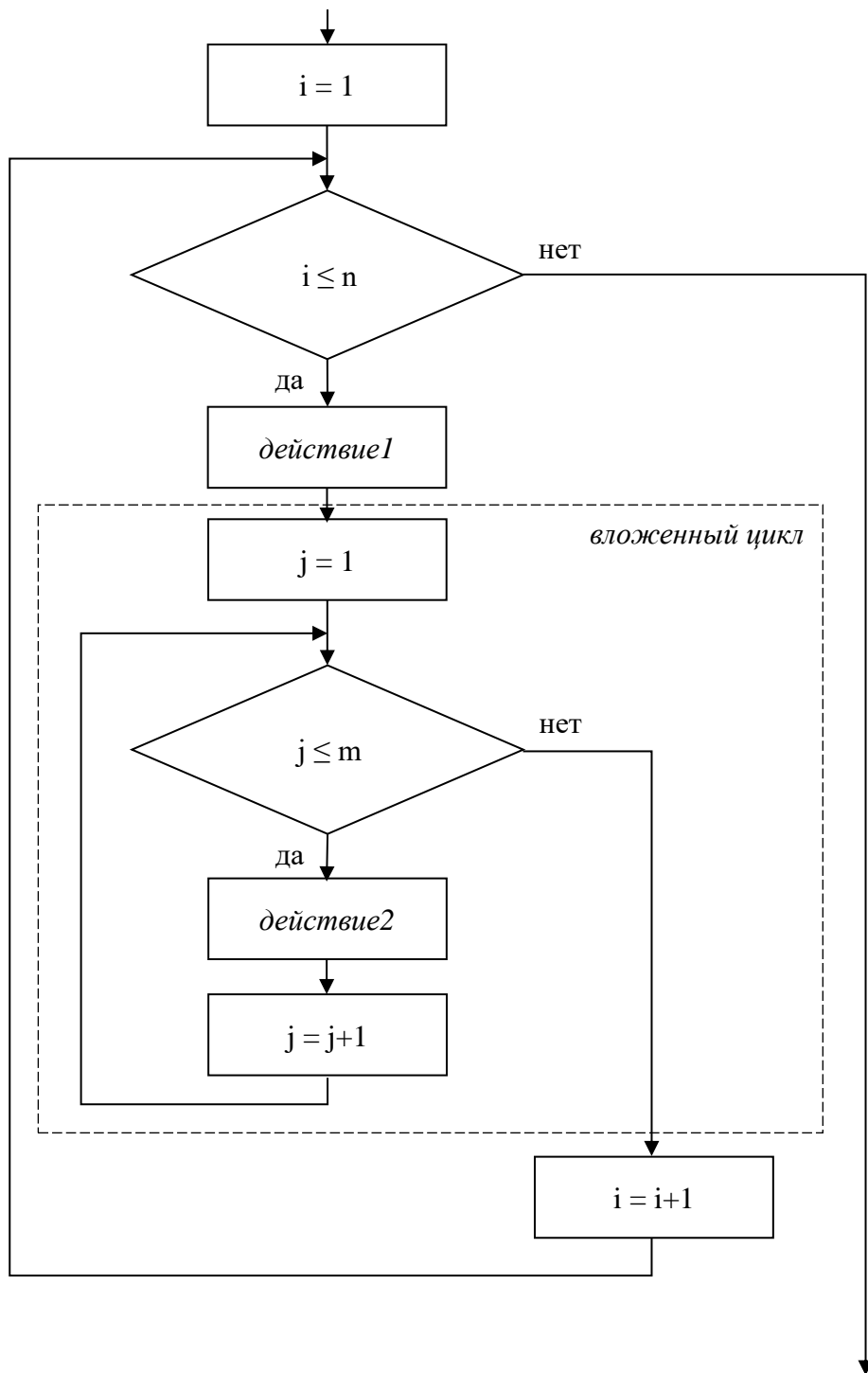
```
for (i = 1; i <= n; i++)  
{  
    действие  
}
```



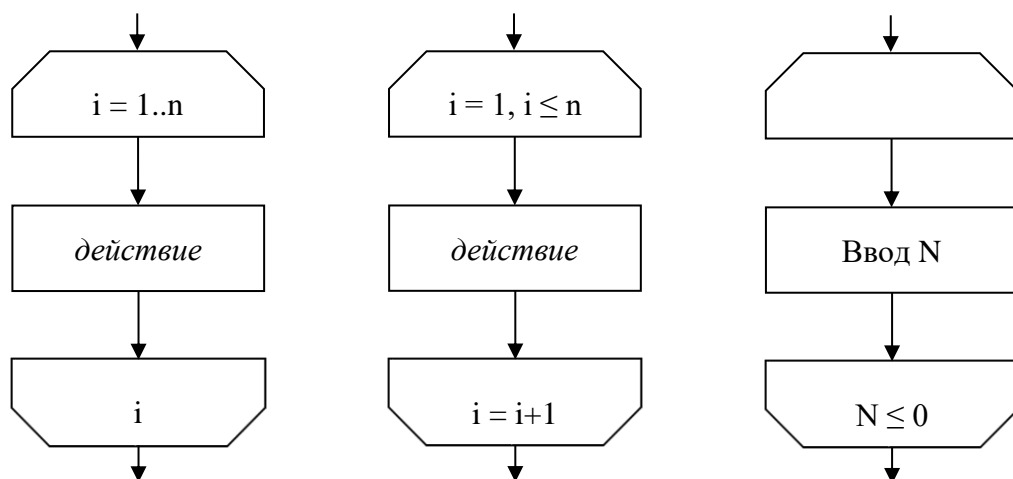
*Действие* – одно или несколько действий (операторов), выполняемых в цикле. Это может быть еще один цикл – вложенный цикл. Вложенный цикл изображается точно также внутри другого цикла вместо блока «действие» (или может представлять собой одно из действий), например:

```
i = 1;  
while (i <= n)  
{  
    действие1  
    j = 1;  
    while (j <= m)  
    {  
        действие2  
        j++;  
    }  
    i++;  
}
```

```
for (i = 1; i <= n; i++)  
{  
    действие1  
    for (j = 1; j <= m; j++)  
    {  
        действие2  
    }  
}
```



ГОСТ также предлагает альтернативное изображение циклов с помощью специальных блоков, в которых указываются специфические действия, выполняемые в начале или в конце итерации, или указывается условие цикла, закон (интервал) изменения переменной – параметры цикла, и т.п. Обозначения в начальном и конечном блоках цикла могут подчеркивать его структуру – с предусловием или постусловием. Если циклы вложенные или следуют друг за другом, то обозначения в начальном и конечном блоках должны показывать, что они относятся к одному циклу, например, указанием одной и той же переменной цикла.



Иногда в литературе встречается обозначение, которое не является стандартным (не предусмотрено ГОСТ). Применение такого изображения цикла **не приветствуется**.

