

Алгоритмы и программирование

Алгоритм - предписание, однозначно задающее процесс преобразования исходной информации в виде последовательности элементарных дискретных шагов, приводящих за конечное число их применений к результату.

Программа - алгоритм решения задачи записанный на понятном машине языке в виде точно определенной последовательности операций - программы для ЭВМ. Составление программ (программирование) обычно производится с помощью промежуточного (алгоритмического) языка.

В алгоритмах и программах для обозначения данных используются **переменные** , которым даются уникальные имена (идентификаторы).

Алгоритмы будем представлять в виде блок-схем, а программы записывать на языке программирования MATLAB

Основные элементы блок-схем



– Начало вычислительного процесса.



– Конец вычислений.



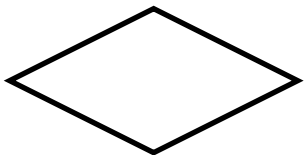
– Блок обмена информацией;
ввод данных и вывод результатов.



– Вычислительный блок;
выполнение операции или группы операций
вычислительного процесса.



– Алгоритмический блок;
использование ранее созданных и от
дельно описанных алгоритмов.

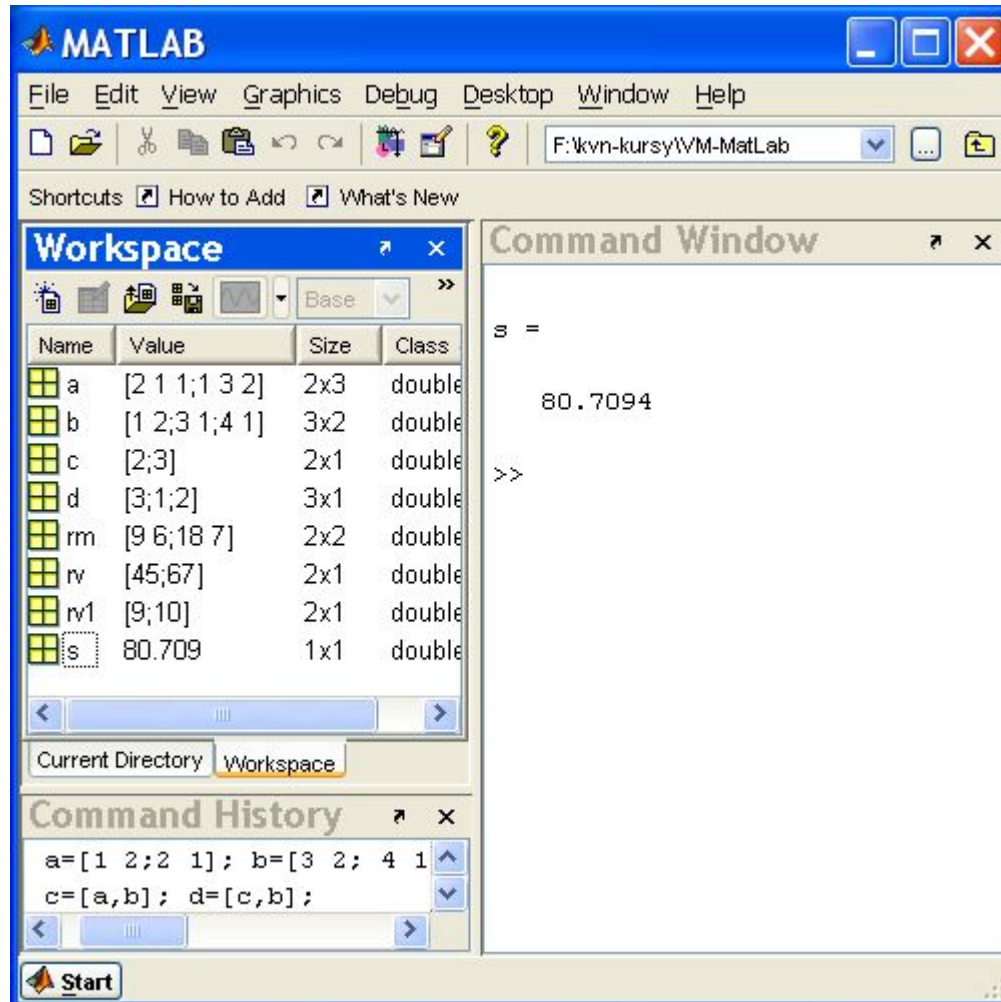


– Логический блок;
выбор направления выполнения алгоритма
в зависимости от условия.



– Циклический блок;
организует многократное выполнение
вложенных блоков.

Программа в MATLAB оформляется в виде М-файла и осуществляет необходимые операции над массивами. **Массив** представляет собой набор переменных с одним именем, одним типом и разными индексами. Каждая переменная называется элементом массива. Количество хранящихся в массиве элементов называется размером массива.



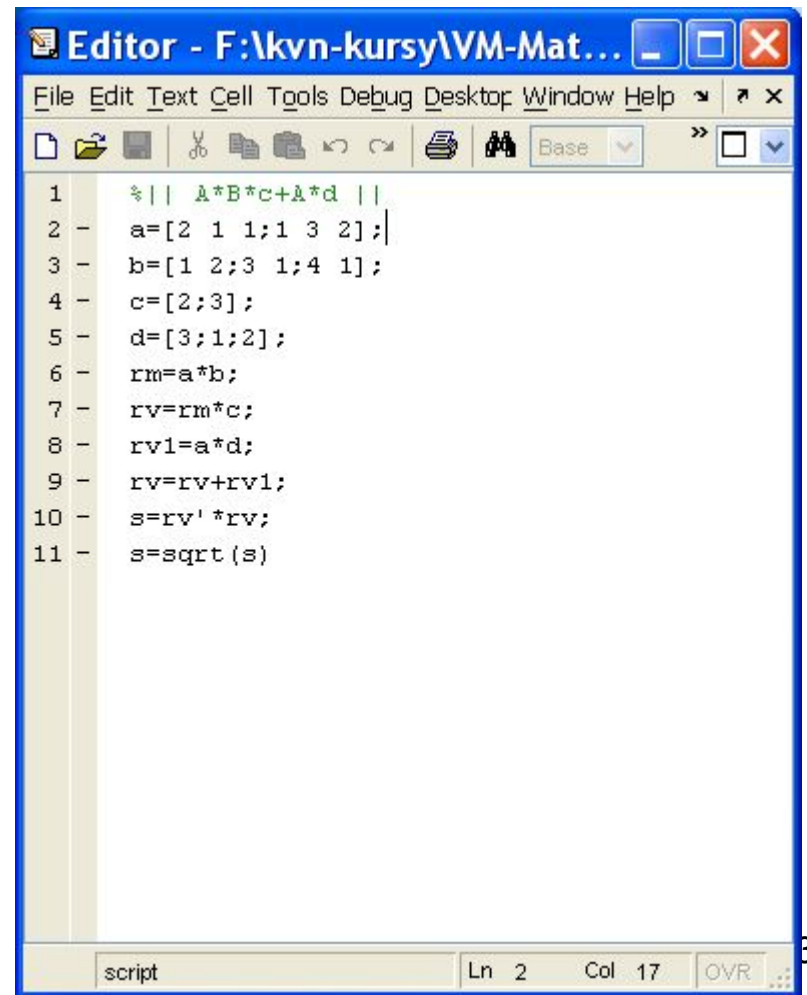
The screenshot shows the MATLAB desktop environment. The main window is titled "MATLAB" and contains several panes:

- Workspace:** A table listing variables and their properties.
- Command Window:** Shows the current command prompt and the result of the last command.
- Command History:** Lists previously executed commands.

Name	Value	Size	Class
a	[2 1 1; 1 3 2]	2x3	double
b	[1 2; 3 1; 4 1]	3x2	double
c	[2; 3]	2x1	double
d	[3; 1; 2]	3x1	double
rm	[9 6; 18 7]	2x2	double
rv	[45; 67]	2x1	double
rv1	[9; 10]	2x1	double
s	80.709	1x1	double

Command Window output:
s =
80.7094
>>

Command History:
a=[1 2; 2 1]; b=[3 2; 4 1]
c=[a,b]; d=[c,b];



The screenshot shows a MATLAB M-file editor window titled "Editor - F:\kvn-kursy\VM-Mat...". The editor contains the following MATLAB code:

```
1 %|| A*B*c+A*d ||  
2 - a=[2 1 1; 1 3 2];  
3 - b=[1 2; 3 1; 4 1];  
4 - c=[2; 3];  
5 - d=[3; 1; 2];  
6 - rm=a*b;  
7 - rv=rm*c;  
8 - rv1=a*d;  
9 - rv=rv+rv1;  
10 - s=rv' *rv;  
11 - s=sqrt(s)
```

The status bar at the bottom indicates the file name "script", line number "Ln 2", column number "Col 17", and the "OVR" (Overwrite) mode.

М-файл сценария Основные свойства

1. Не имеет входных и выходных параметров.
2. Работает с данными из рабочей области.
3. В процессе выполнения не компилируется.
4. Представляет собой последовательность операций.

```
%|| a*d*c+a*d ||  
a=[2 1 1;1 3 2];  
b=[1 2;3 1;4 1];  
c=[2;3];  
d=[3;1;2];  
rm=a*b;  
rv=rm*c;  
rv1=a*d;  
rv=rv+rv1;  
s=rv'*rv;  
s=sqrt(s)
```

М-файл функция

Общий вид

function <список выходных параметров>=<имя функции>(список входных параметров)
<операторы>
<список выходных параметров>=<выражение>

Основные свойства

1. Начинается с ключевого слова **function**
2. Результат выполнения присваивается имени функции

```
function f=Fxy(x)  
f=(x(1)^2+x(2)^2-3)^2+(x(1)^2+x(2)^2-2*x(1)-3)^2+1;
```

Обращение к функции

(может содержать файл функция либо файл сценария)

```
x=[2 3];  
y=Fxy(x);
```

Структуры алгоритмов и их программирование на VBA

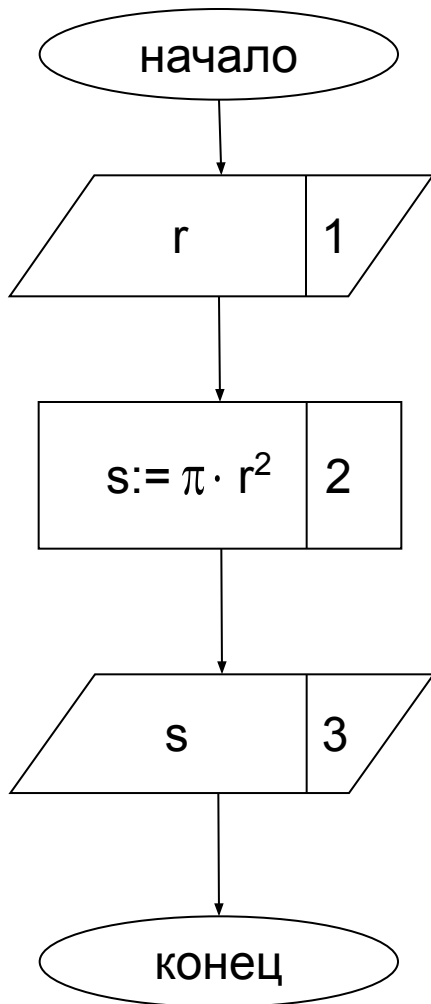
Любой алгоритм можно представить как совокупность некоторых базовых, т.е. основных структур таких как:

следование, ветвление и цикл.

СЛЕДОВАНИЕ

Последовательное исполнение блоков один за другим.

Вычислить площадь круга. $s = \pi \cdot r^2$

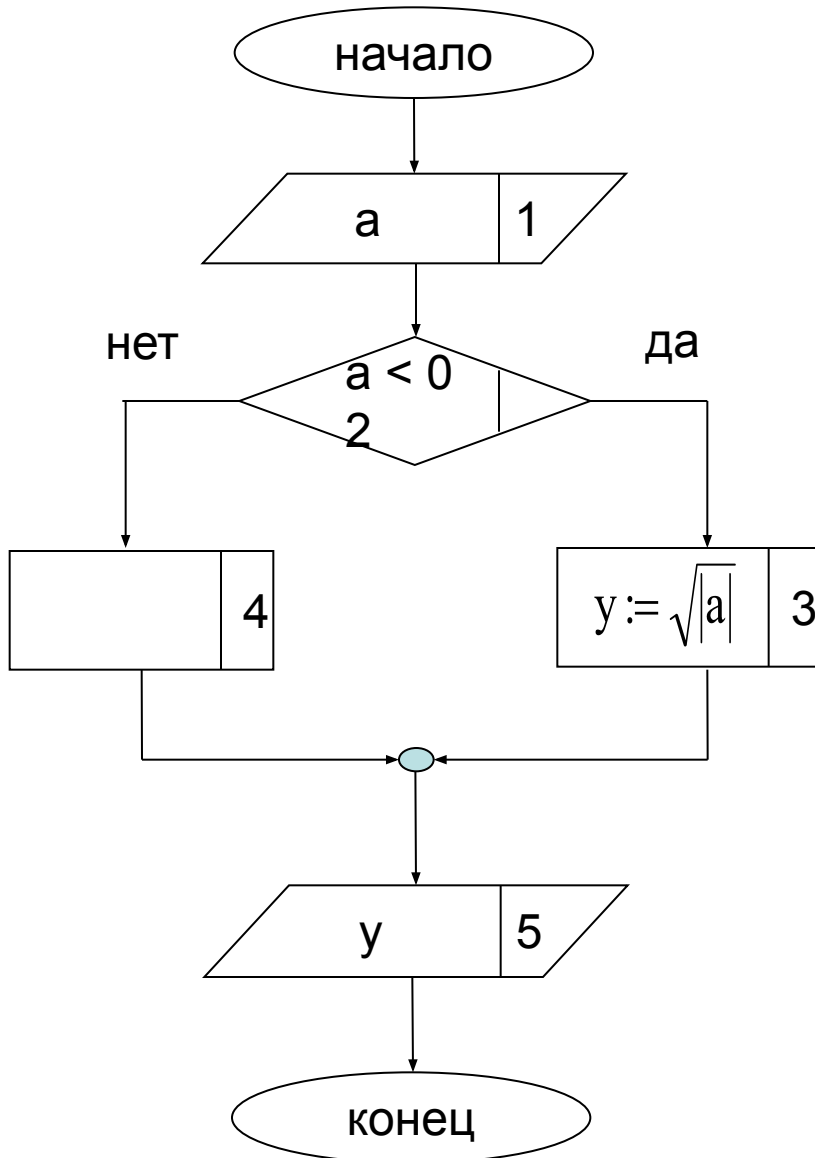


```
clc
r=input('r=');
s=pi*r^2;
str=sprintf('s=%.4e',s);
disp(str)
```

ВЕТВЛЕНИЕ

Составить блок-схему алгоритма и программу вычисления функции:

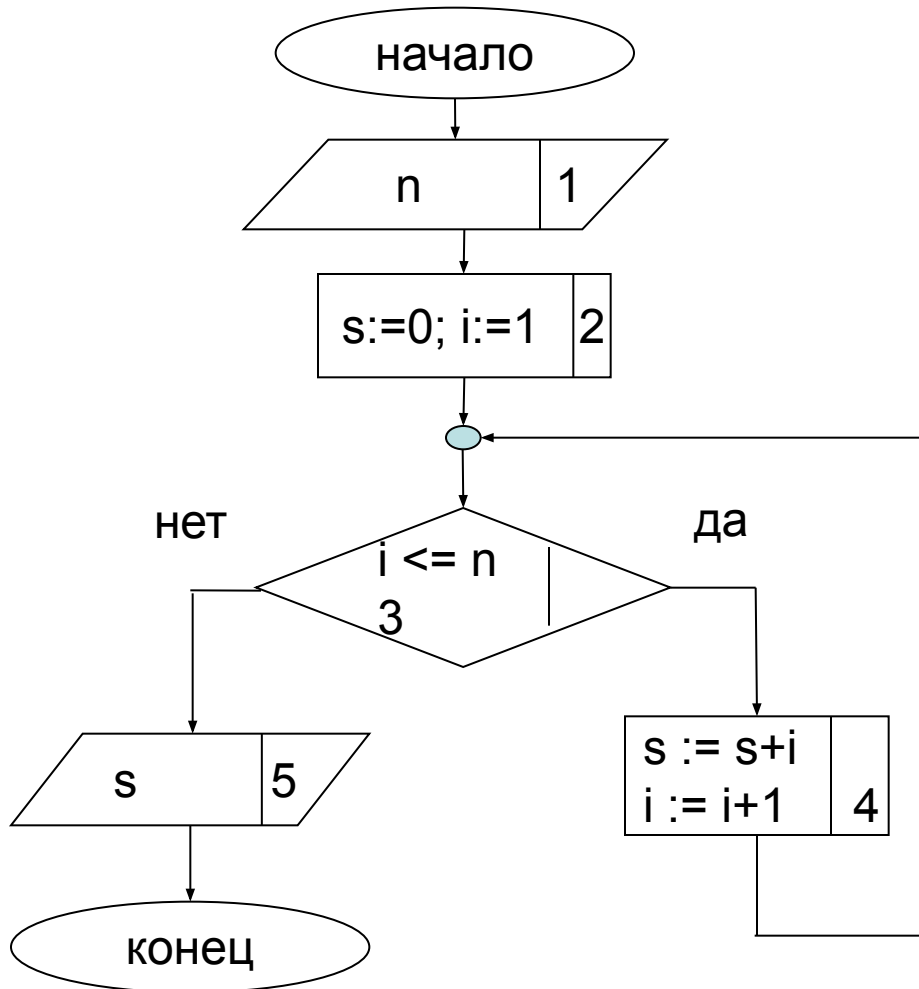
$$y = \begin{cases} \sqrt{|a|}, & \text{если } a < 0 \\ \sqrt{a}, & \text{если } a \geq 0 \end{cases}$$



```
clc
a=input(' введите a=');
if a<0
    y=sqrt(abs(a));
else
    y=sqrt(a);
end
str=sprintf('y=%.4e',y);
disp(str)
```

Составить блок-схему алгоритма и программу вычисления суммы n чисел натурального ряда. $S=1+2+3+\dots+n$

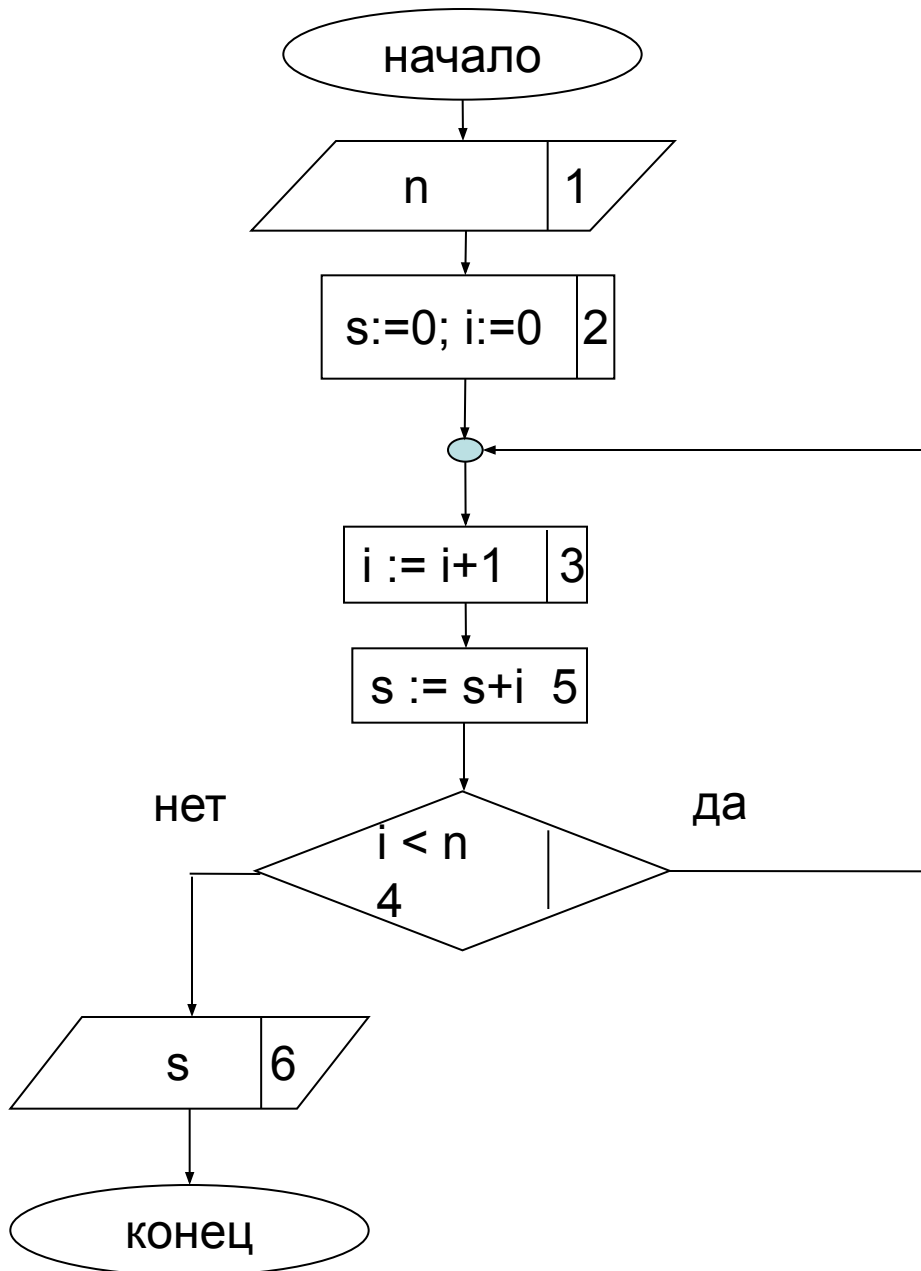
ЦИКЛ С ПРЕДУСЛОВИЕМ



```
clc
n=input('n=');
s=0;
i=1;
while i<=n
    s=s+i;
    i=i+1;
end
str=sprintf('s=%5g',s);
disp(str)
```

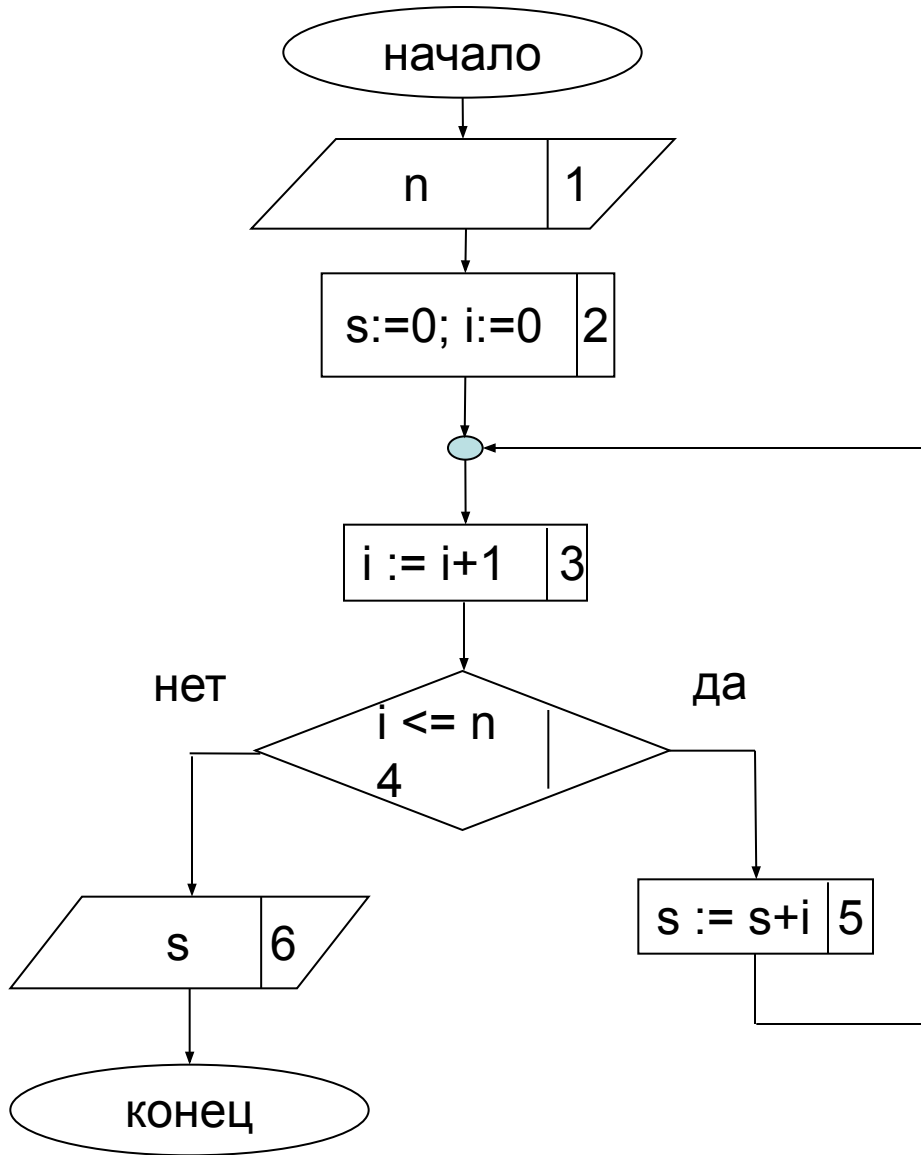
Инструкция **break**– Выход из цикла в произвольном месте

ЦИКЛ С ПОСТУСЛОВИЕМ



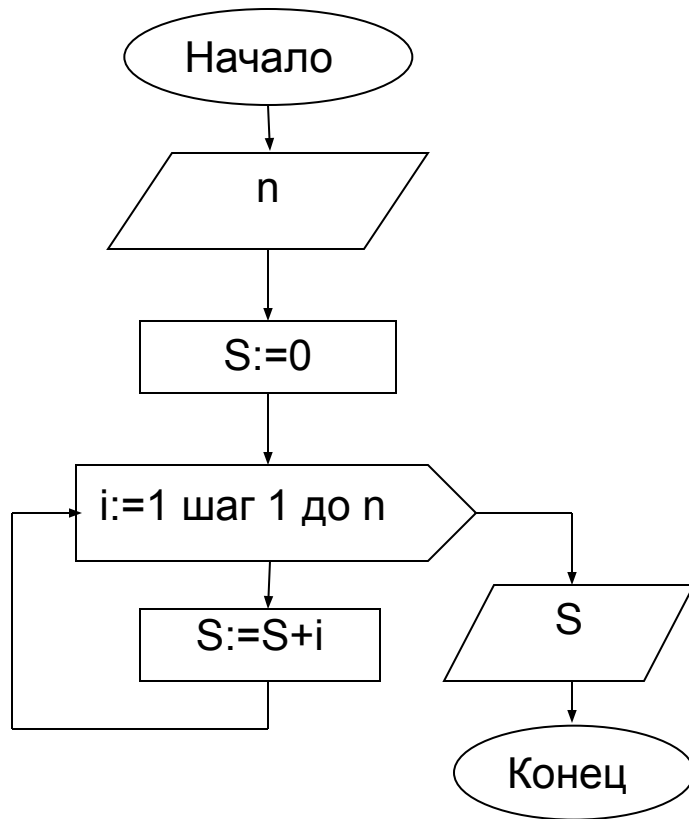
```
clc
n=input('n=');
s=0;
i=0;
while (1>0)
    i=i+1;
    s=s+i;
    if i >= n break; end
end
str=sprintf('s=%5g',s);
disp(str)
```

ЦИКЛ БЕЗУСЛОВИЯ



```
clc
n=input('n=');
s=0;
i=0;
while (1>0)
    i=i+1;
    if i > n break; end
    s=s+i;
end
str=sprintf('s=%5g',s);
disp(str)
```

ЦИКЛ С ПАРАМЕТРОМ



```
clc  
n=input('n=');  
s=0;  
for i=1:n  
    s=s+i;  
end  
str=sprintf('s=%5g',s);  
disp(str)
```

Инструкция **break**– Выход из цикла в произвольном месте