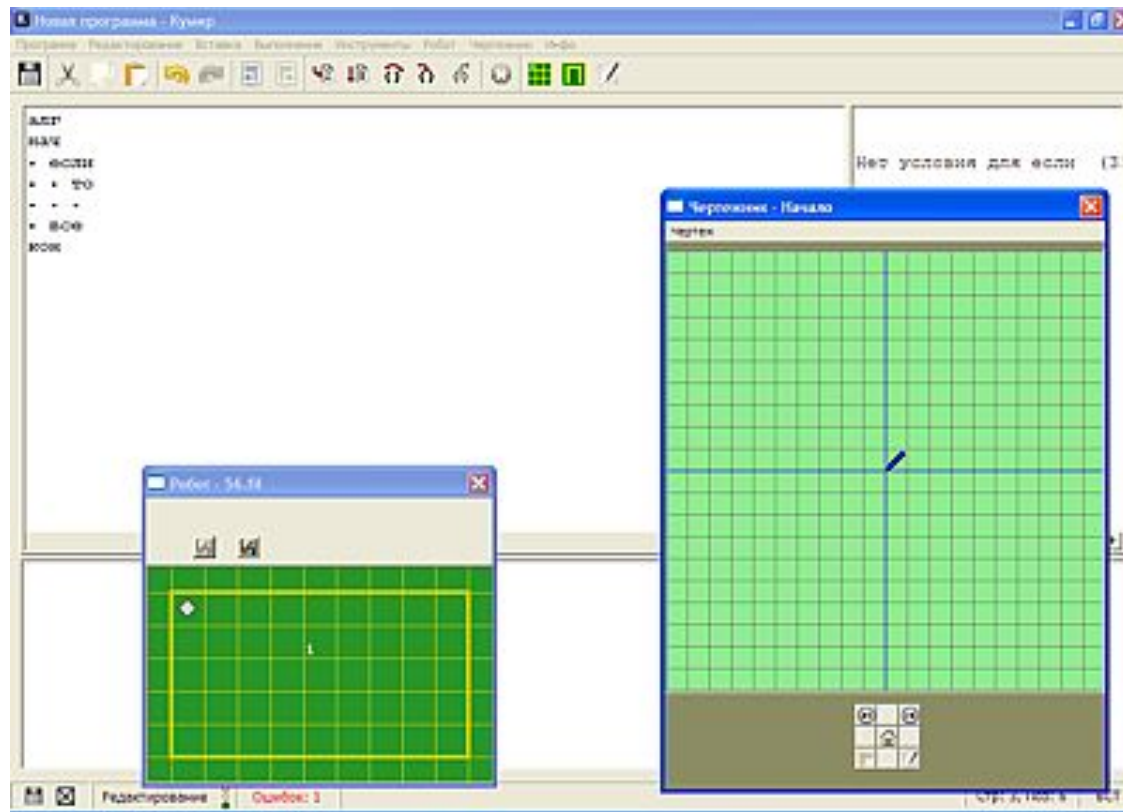


**алгоритмы**

# <http://www.niisi.ru/kumir/>

**КуМир** (Комплект Учебных МИРов) - система программирования, предназначенная для поддержки начальных курсов информатики и программирования в средней и высшей школе.



## КОМАНДЫ РОБОТА

вверх            вниз  
вправо          влево

закрасить

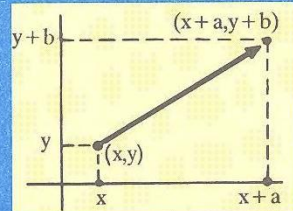
ЛОГ сверху стена            ЛОГ сверху свободно  
ЛОГ снизу стена              ЛОГ снизу свободно  
ЛОГ справа стена            ЛОГ справа свободно  
ЛОГ слева стена              ЛОГ слева свободно  
ЛОГ клетка закрашена      ЛОГ клетка не закрашена

ВЕЩ температура

ВЕЩ радиация

## КОМАНДЫ ЧЕРТЕЖНИКА

поднять перо  
опустить перо  
сместиться в точку( арг вещ  $x, y$  )  
сместиться на вектор( арг вещ  $a, b$  )



## ОБЩИЙ ВИД АЛГОРИТМА

алг имя( аргументы и результаты )  
дано условия применимости алгоритма  
надо цель выполнения алгоритма  
нач описание промежуточных величин  
тело алгоритма (последовательность команд)  
кон

## КОМАНДЫ АЛГОРИТМИЧЕСКОГО ЯЗЫКА

нц число повторений раз

| тело цикла (последовательность команд)

кц

нц пока условие

| тело цикла (последовательность команд)

кц

нц для  $i$  от  $i_1$  до  $i_2$

| тело цикла (последовательность команд)

кц

если условие

| то серия 1

| иначе серия 2

все

если условие

| то серия 1

все

выбор

| при условие 1 : серия 1

| при условие 2 : серия 2

...

| при условие  $n$  : серия  $n$

| иначе серия  $n+1$

все

выбор

| при условие 1 : серия 1

| при условие 2 : серия 2

...

| при условие  $n$  : серия  $n$

все

утв условие

ввод имена величин

вывод тексты, имена величин, выражения, нс

вызов:

имя алгоритма( аргументы и имена результатов )

присваивание:

имя величины := выражение

## ТИПЫ ВЕЛИЧИН

целые	<u>цел</u>
вещественные	<u>вещ</u>
логические	<u>лог</u>
символьные	<u>сим</u>
литерные	<u>лит</u>

### Таблицы:

целые	<u>цел таб</u>
вещественные	<u>вещ таб</u>
логические	<u>лог таб</u>
символьные	<u>сим таб</u>

Пример описания: цел i, j, лит t, вещ таб a [1:50]

## ВИДЫ ВЕЛИЧИН

аргументы ( <u>арг</u> )	- описываются в заголовке алгоритма
результаты ( <u>рез</u> )	- описываются в заголовке алгоритма
значения функций ( <u>знач</u> )	- описываются указанием типа перед именем алгоритма-функции
промежуточные	- описываются в строке <u>нач</u> алгоритма
общие	- описываются после строки <u>исп</u> исполнителя

## ОБЩИЙ ВИД ИСПОЛНИТЕЛЯ

исп имя

описание общих величин исполнителя  
команды для задания начальных значений общих величин  
алгоритмы исполнителя

кон

## ОБОЗНАЧЕНИЯ ЗНАКОВ ОПЕРАЦИЙ И СТАНДАРТНЫХ ФУНКЦИЙ

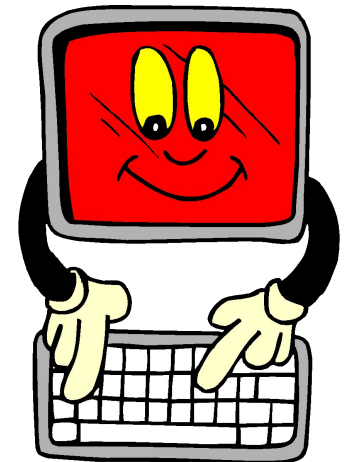
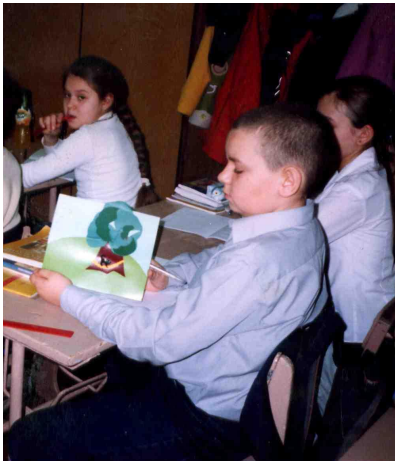
Название операции или функции		Форма записи
сложение		$x + y$
вычитание		$x - y$
умножение		$x * y$
деление		$x / y$
возведение в степень		$x ** y$
корень квадратный		sqrt(x)
абсолютная величина		abs(x)
знак числа (-1, 0 или 1)		sign(x)
синус	sin x	sin(x)
косинус	cos x	cos(x)
тангенс	tg x	tg (x)
котангенс	ctg x	ctg(x)
арксинус	arcsin x	arcsin(x)
арккосинус	arccos x	arccos(x)
арктангенс	arctg x	arctg (x)
арккотангенс	arcctg x	arcctg(x)
натуральный логарифм	ln x	ln(x)
десятичный логарифм	lg x	lg(x)
степень числа e ( $e \approx 2.718281$ )	$e^x$	exp(x)
минимум из чисел x и y		min(x,y)
максимум из чисел x и y		max(x,y)
остаток от деления x на y (x, y – целые)		mod(x,y)
частное от деления x на y (x, y – целые)		div(x,y)
целая часть числа x		int(x)
случайное число в диапазоне от 0 до x		rnd(x)

# Алгоритм

- последовательность указаний (команд) исполнителю, выполнив которую, он достигает поставленной цели или решает определенную задачу.

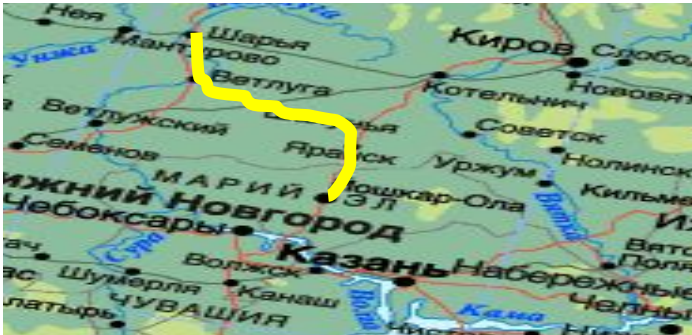
# Исполнитель

- это система, понимающая определенные команды и обладающая инструментами для выполнения команд.



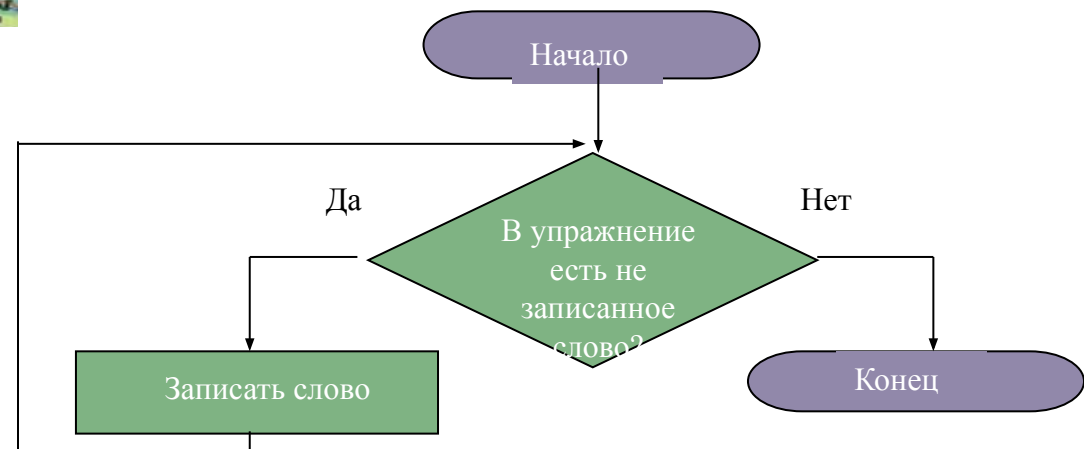
# Как задать алгоритм?

В виде инструкций, рецептов



С помощью рисунков, схем.  
Например, путь,  
обозначенный на карте.

Блок-схема — запись алгоритма в виде последовательности специальных графических блоков-обозначений.



В зависимости от того, каков порядок использования команд, можно выделить три типа алгоритмов:

**Линейный**

**Условный (разветвляющийся)**

**Циклический**





Алгоритм решения задачи называется *линейным*,  
если исполнитель все команды алгоритма  
выполняет одну за другой в порядке их записи.

Задание.

Разберите слово по  
составу.

## Алг. Состав слова

### Нач.

1. Выделяем окончание слова
2. Выделяем основу слова
3. Выделяем корень
4. Выделяем приставку
5. Выделяем суффикс.

### Кон.

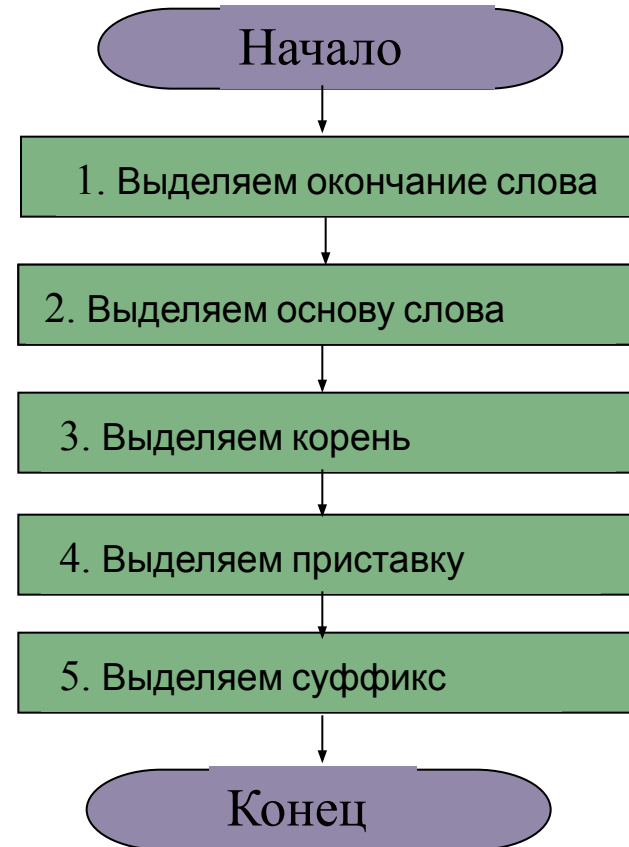
# Построим блок-схему для линейного алгоритма

## Алг. Состав слова

**Нач.**

1. Выделяем окончание слова
2. Выделяем основу слова
3. Выделяем корень
4. Выделяем приставку
5. Выделяем суффикс.

**Кон.**



Алгоритм называется *разветвляющимся*, если после проверки условия в разных ситуациях исполняются разные наборы команд.

Составим алгоритм правописания приставок на «з», «с».

Например:

Разговор (пишем «з»)

Рассказ (пишем «с»)

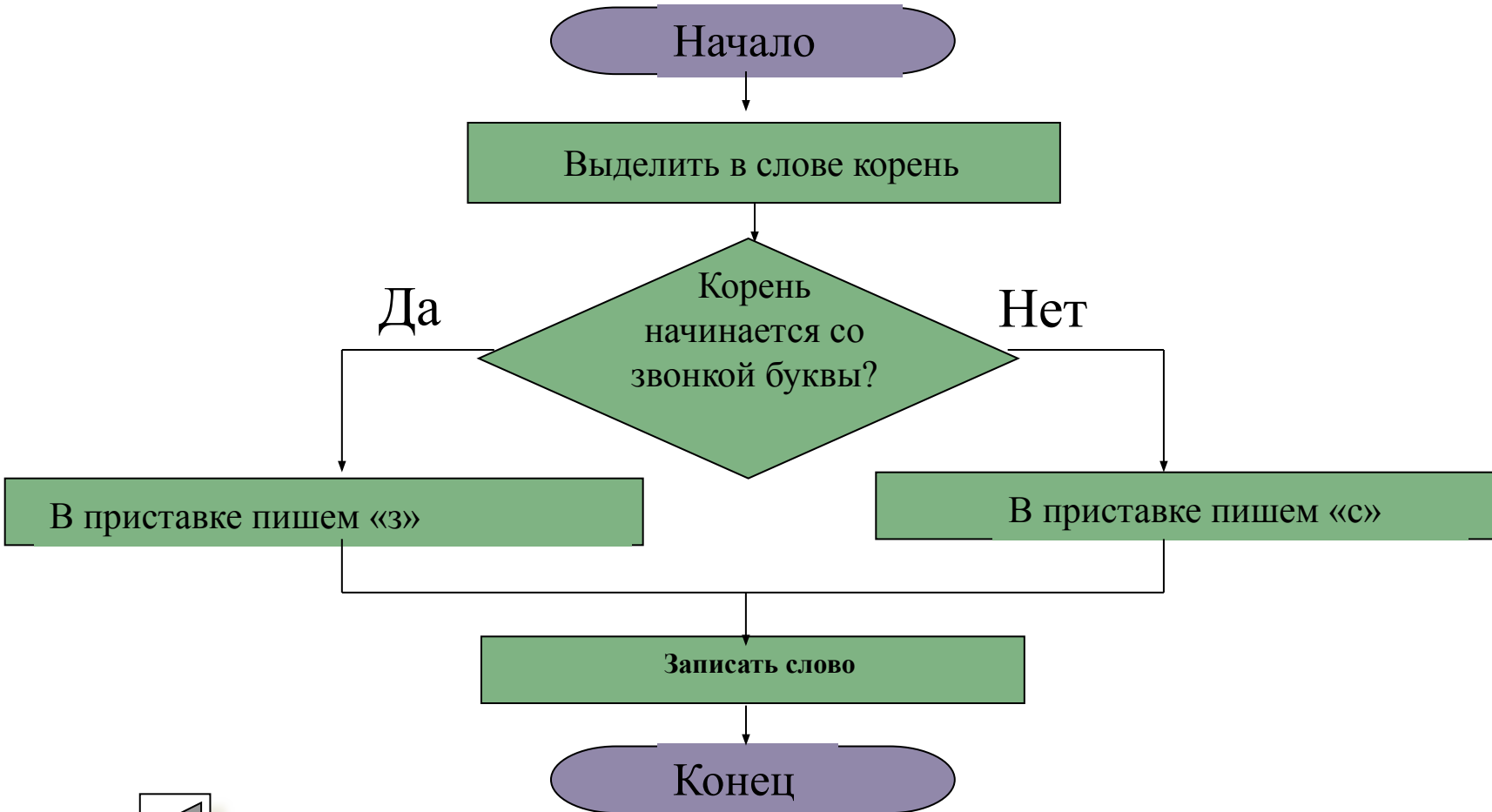
## Алг. Правописание

### Нач.

1. Выделяем приставку в слове.
2. Если после приставки звонкий звук, перейти к пункту 3, в противном случае — к пункту 4
3. В приставке пишем «з». Перейти к пункту 5
4. В приставке пишем «с».
5. Записать слово

### Кон.

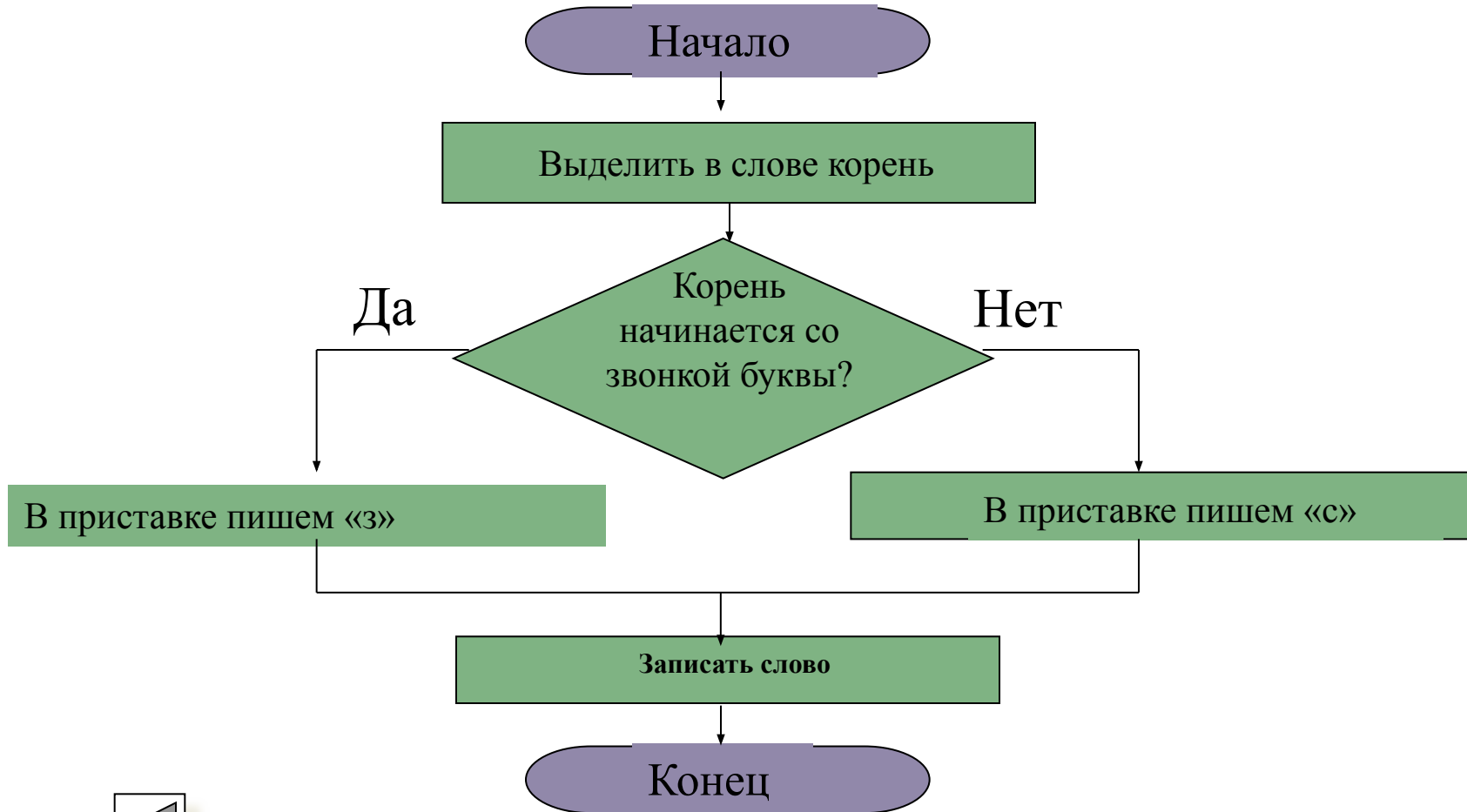
# Построим блок-схему для разветвляющегося алгоритма



Вставить букву в слова:

Бе\_заботный

Ра\_писание



Если в алгоритме одну и ту же команду приходится записывать несколько раз, то для сокращения алгоритма используется специальная команда *повторения*.

Алгоритм, который содержит команду повторения, называют *алгоритмом с повторением или циклическим*.

Составим алгоритм задачи:

***Как нужно запоминать определения терминов и закономерностей.***

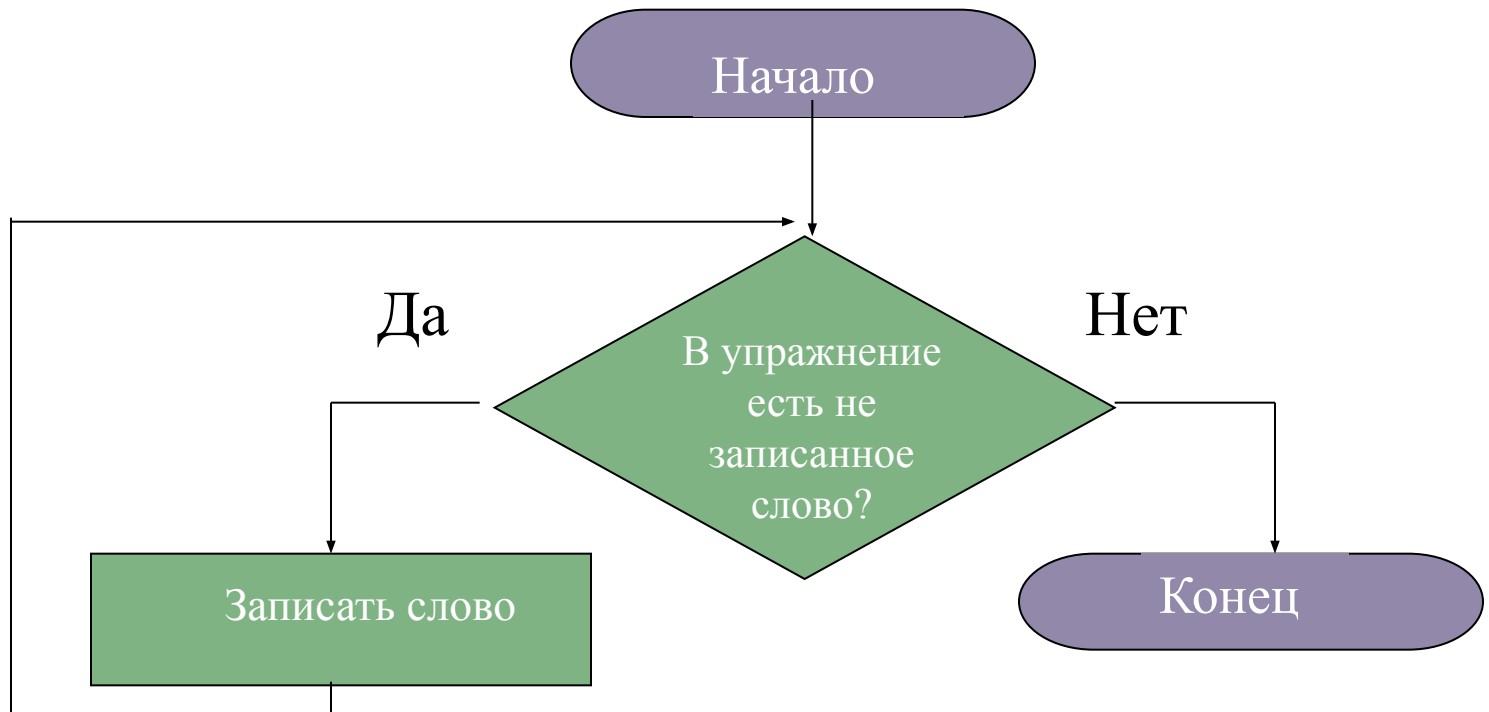
1. Запишите определения термина в тетрадь.
2. Прочтите записанное.
3. Изобразите определение в виде рисунка, схемы, слов и сравните с записью.
4. Прочтите изображение.
5. Повторите два-три раза весь цикл.

# Построим блок-схему для алгоритма с повторением



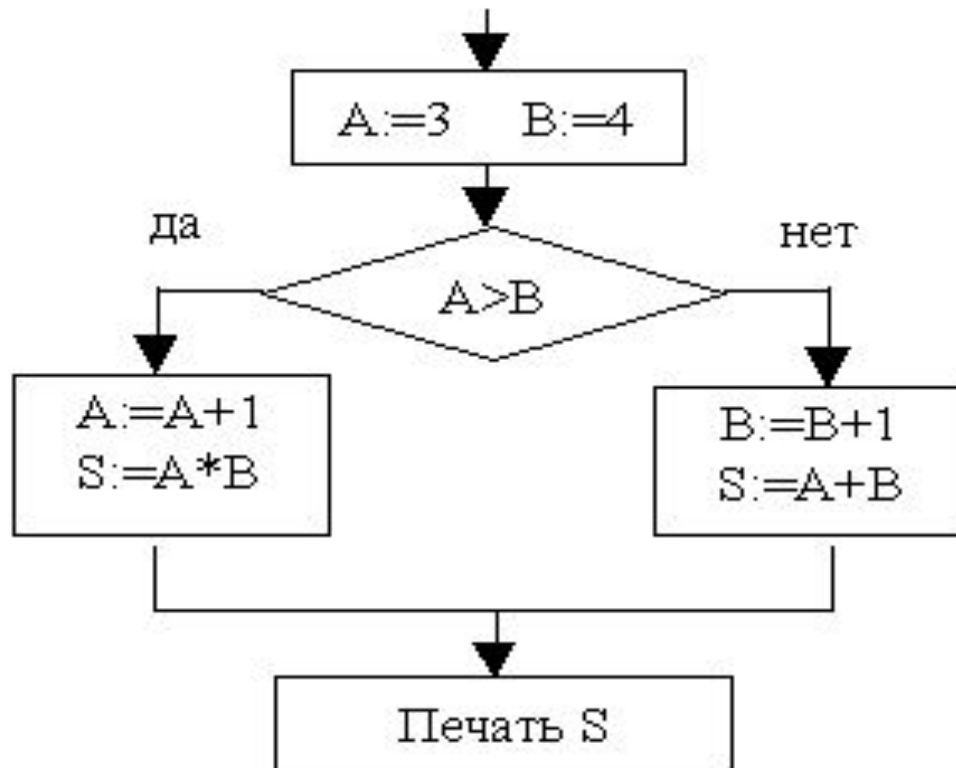
# Построим блок-схему для алгоритма с повторением

Переписать текст упражнения в тетрадь.



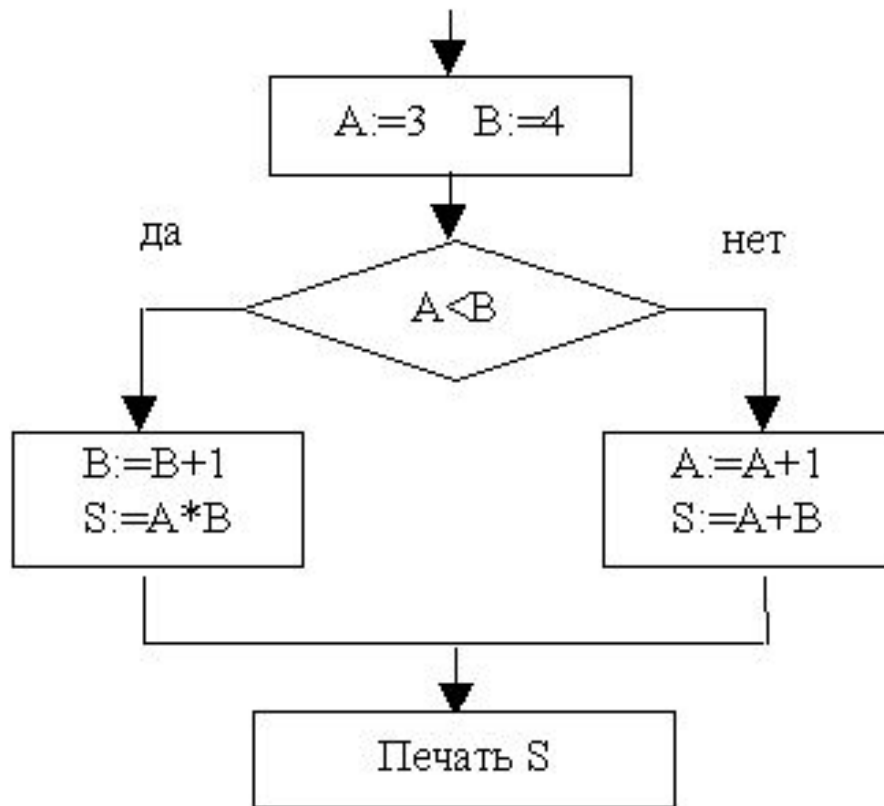


Фрагмент алгоритма изображен в виде блок-схемы. Определите, какое значение переменной  $S$  будет напечатано в результате выполнения алгоритма.



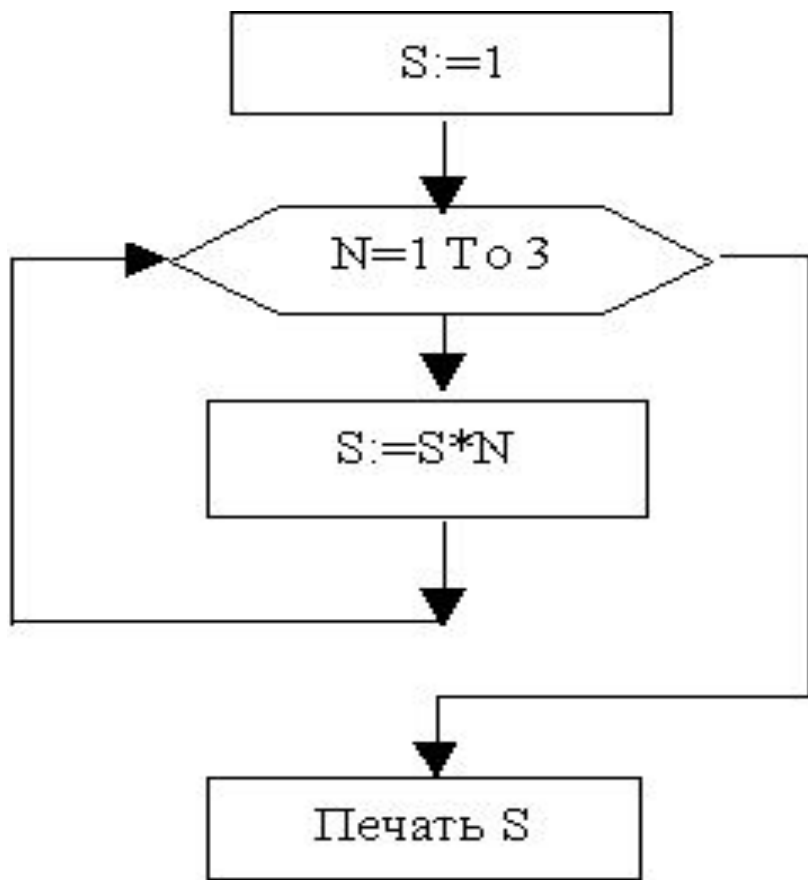
*Ответ:8*

Фрагмент алгоритма изображен в виде блок-схемы.  
Определите, какое значение переменной  $S$  будет напечатано в результате выполнения алгоритма.



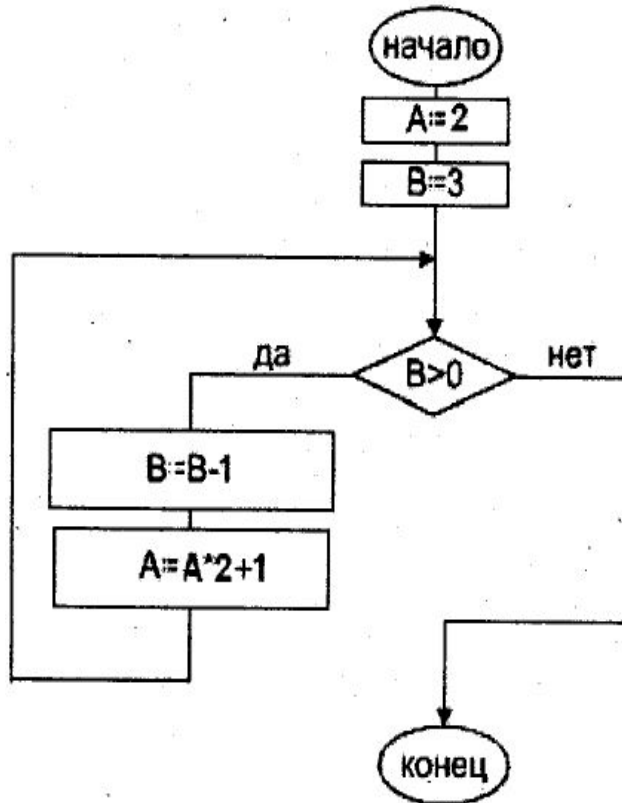
Ответ:15

Фрагмент алгоритма изображен в виде блок-схемы. Определите, какое значение переменной  $S$  будет напечатано в результате выполнения алгоритма.



Ответ: 6

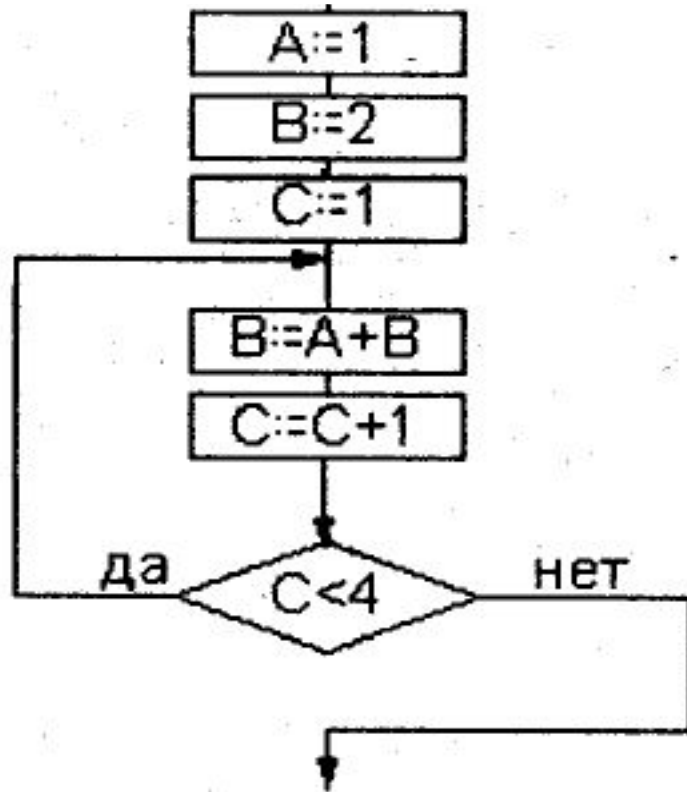
Определите значение переменных А и В после выполнения следующего фрагмента алгоритма:



A	B
2	-
2	3
3 > 0 – да	
1	2=3-1
5=2*2+1	2
2 > 0 – да	
5	1=2-1
11=5*2+1	1
1 > 0 – да	
11	0=1-1
23=11*2+1	0
0 > 0 – нет	

Ответ: A=23, B=0

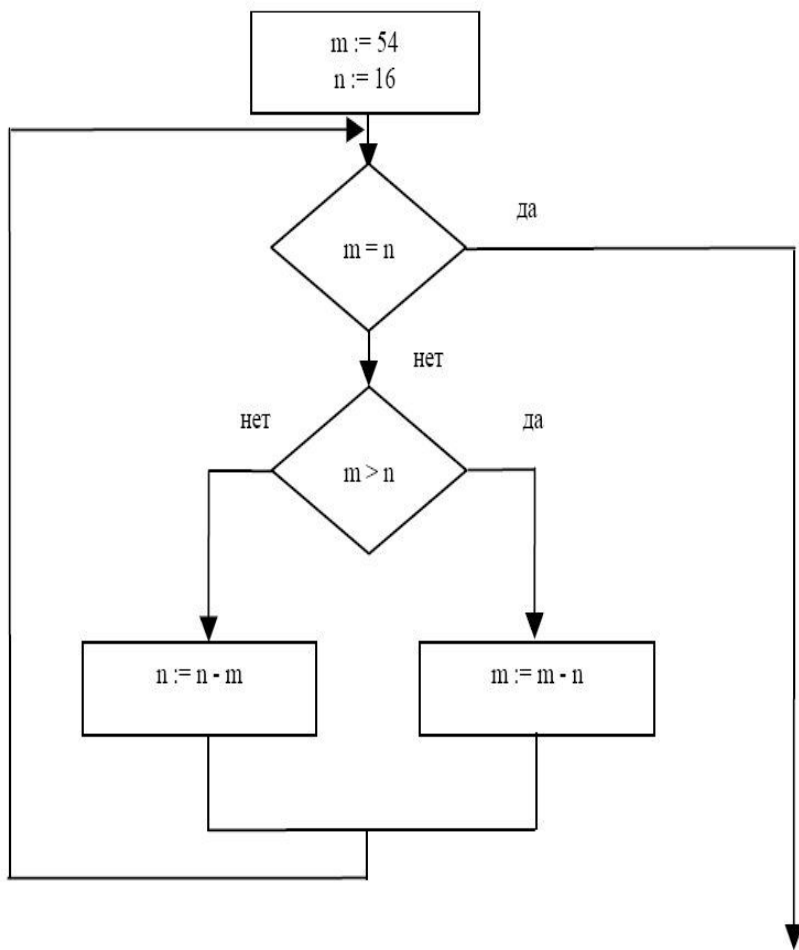
Определите значение переменной В после выполнения следующего фрагмента алгоритма:



A	B	C
1	-	-
1	2	-
1	2	1
1	$3=1+2$	1
1	3	$2=1+1$
$2 < 4$ – да		
1	$4=1+3$	2
1	4	$3=2+1$
$3 < 4$ – да		
1	$5=1+4$	3
1	5	$4=3+1$
$4 < 4$ – нет		

Ответ: 5

# Определите значение переменной $m$ после выполнения фрагмента алгоритма.



Примечание: знаком := обозначена операция присваивания.

m	n
54	16
54=16 – нет	
54>16 - да	
38	16
38=16 – нет	
38>16 - да	
22	16
22=16 - нет	
22 >16 - да	
6	16
6 = 16 - нет	
6 > 16 – нет	
6	10
6 = 10 – нет	
6 > 10 – нет	
6	4
6 = 4 – нет	
6 > 4 – да	
2	4
2=4 – нет	
2>4 – нет	
<b>Ответ: 2</b>	
2=2 – да	

Исполнитель Вычислитель работает с тремя командами, которым присвоены номера:

1. Умножить на два
2. Вычесть один
3. Прибавить три

Выполняя первую из них, Вычислитель удваивает число на экране, вторую- уменьшает число на экране, третью- увеличивает число на три. Запишите порядок команд в программе получения числа 29 из 6, содержащей не более 5-ти команд, указывая лишь номера команд.

Например, последовательность 21321 соответствует программе:

Вычесть один

Умножить на два

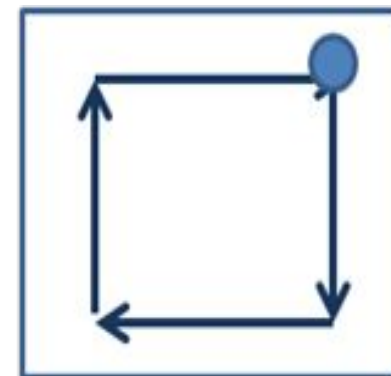
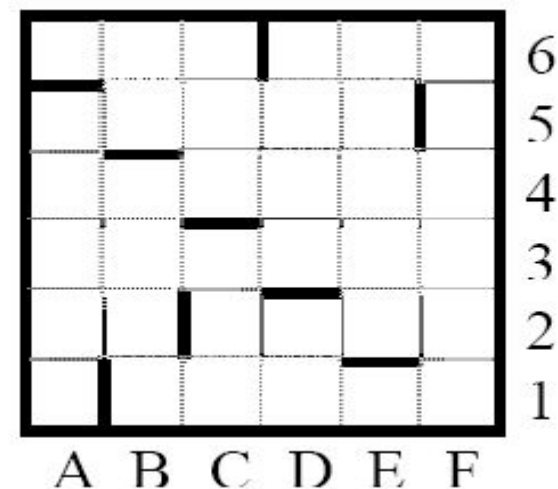
Прибавить три

Вычесть один

Умножить на два,-

которая преобразует число 2 в 8.

- Система команд исполнителя РОБОТ, «живущего» в прямоугольном лабиринте на клетчатой плоскости:
- |               |             |              |  |
|---------------|-------------|--------------|--|
| <b>вверх</b>  | <b>вниз</b> | <b>влево</b> |  |
| <b>вправо</b> |             |              |  |
- При выполнении любой из этих команд РОБОТ перемещается на одну клетку соответственно: вверх  $\uparrow$ , вниз  $\downarrow$ , влево  $\leftarrow$ , вправо  $\rightarrow$ .
- Четыре команды проверяют истинность условия отсутствия стены у каждой стороны той клетки, где находится РОБОТ:
- |                        |                        |              |  |
|------------------------|------------------------|--------------|--|
| <b>сверху свободно</b> | <b>снизу свободно</b>  | <b>слева</b> |  |
| <b>свободно</b>        | <b>справа свободно</b> |              |  |
- Цикл ПОКА *< условие >* команда
- выполняется, пока условие истинно, иначе происходит переход на следующую строку.
- Сколько клеток приведенного лабиринта соответствуют требованию, что, выполнив предложенную ниже программу, РОБОТ остановится в той же клетке, с которой он начал движение?
- НАЧАЛО
- ПОКА **< снизу свободно >** вниз
- ПОКА **< слева свободно >** влево
- ПОКА **< сверху свободно >** вверх
- ПОКА **< справа свободно >** вправо
- КОНЕЦ
- 1) 1    2) 2    3) 3    4) 0





# Дома:

