

Основные типы

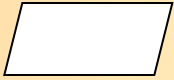
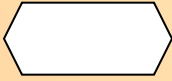
алгоритмических структур.

# Виды алгоритмических структур:

- блок-схема.
- линейный алгоритм.
- алгоритмическая структура «ветвление».
- алгоритмическая структура «выбор».
- алгоритмическая структура «цикл».
- виды циклов.



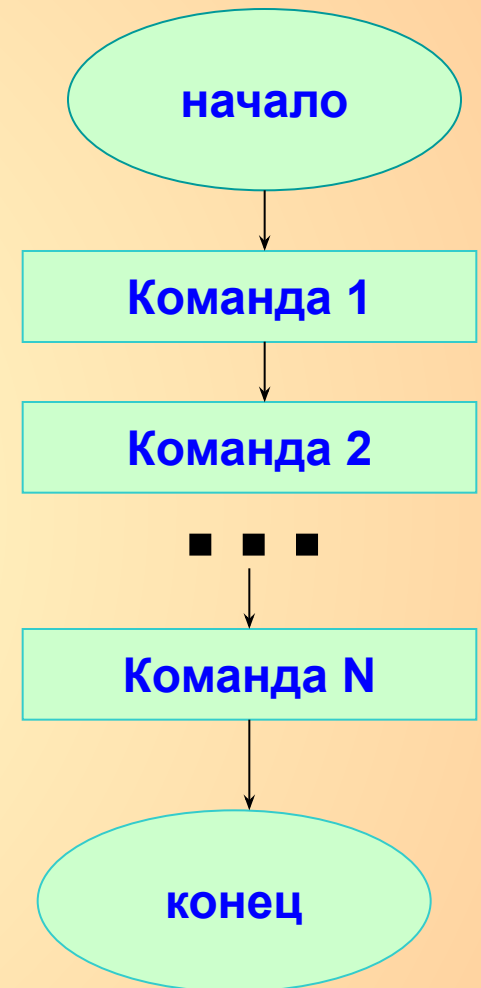
# При записи алгоритмов часто используют блок-схемы:

|                                                                                     |                                                |
|-------------------------------------------------------------------------------------|------------------------------------------------|
|    | Блок, обозначающий начало или конец алгоритма. |
|    | Блок ввода или вывода информации.              |
|    | Блок проверки условия.                         |
|   | Вычислительный блок.                           |
|  | Циклический процесс.                           |

# линейный алгоритм.



Алгоритм, в котором команды выполняются последовательно одна за другой, называется линейным алгоритмом.



# алгоритмическая структура «ВЕТВЛЕНИЕ».

☐ В алгоритмической структуре «ветвление» та или иная серия команд выполняется в зависимости от истинности условия.

☐ Будем называть условием высказывание, которое может быть либо истинным, либо ложным.

Условные выражения могут быть простыми и сложными. Простое условие включает в себя два числа, две переменных или два *арифметических выражения*, которые сравниваются между собой с использованием операций сравнения (равно, больше, меньше и пр.).

Например:  $5 > 3$ ,  $2 * 8 = 4 * 4$  и т. д.

Сложное условие — это последовательность простых условий, объединенных между собой знаками логических операций.

Например,  $5 > 3$  And  $2 * 8 = 4 * 4$ .

# алгоритмическая структура «ветвление».

| VISUAL BASIC                                                                                                                                                                                                                           | Блок-схема                                                                                                                                                                                                                                                                                  |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>1. Полная форма ветвления</p> <pre>IF &lt;условие&gt;<br/>THEN &lt;действие1&gt;<br/>ELSE &lt;действие2&gt;<br/>END IF</pre> <p>2. Неполная форма ветвления</p> <pre>IF &lt;условие&gt;<br/>THEN &lt;действие1&gt;<br/>END IF</pre> | <pre>graph TD; Entry1(( )) --&gt; Cond1{усло<br/>вие}; Cond1 -- да --&gt; Act1[действие1]; Cond1 -- нет --&gt; Act2[действие2]; Act1 --&gt; Exit1(( )); Act2 --&gt; Exit1; Entry2(( )) --&gt; Cond2{усло<br/>вие}; Cond2 -- да --&gt; Act3[действие1]; Cond2 -- нет --&gt; Exit2(( ))</pre> |

# алгоритмическая структура «выбор».



В алгоритмической структуре «выбор» выполняется одна из нескольких последовательностей команд при истинности соответствующего условия

# алгоритмическая структура «выбор».

## VISUAL BASIC

**Select Case** Выражение

**Case** Условие 1

Серия 1

**Case** Условие 2

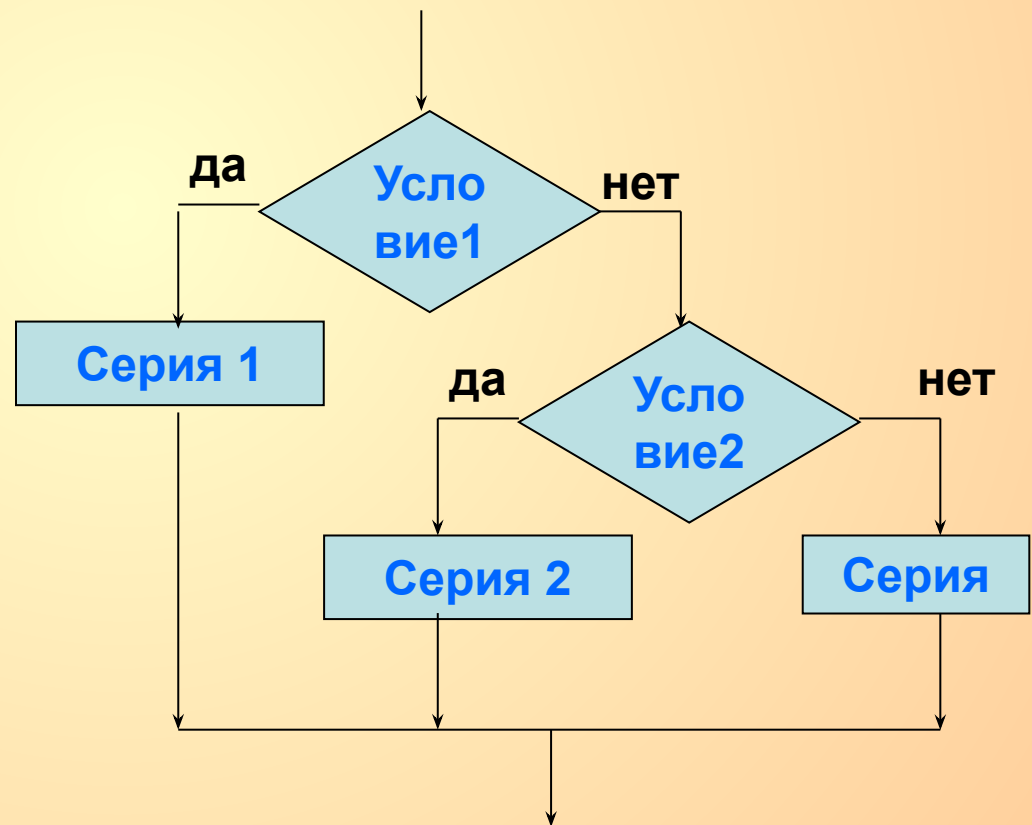
Серия 2

**Case** Else

Серия

**End Select**

## Блок-схема



# алгоритмическая структура «ЦИКЛ».

 В алгоритмической структуре «цикл» серия команд (тело цикла) выполняется многократно

Циклические алгоритмические структуры бывают двух типов:

- циклы со счетчиком, в которых тело цикла выполняется определенное количество раз;
- циклы с условием, в которых тело цикла выполняется, пока условие истинно

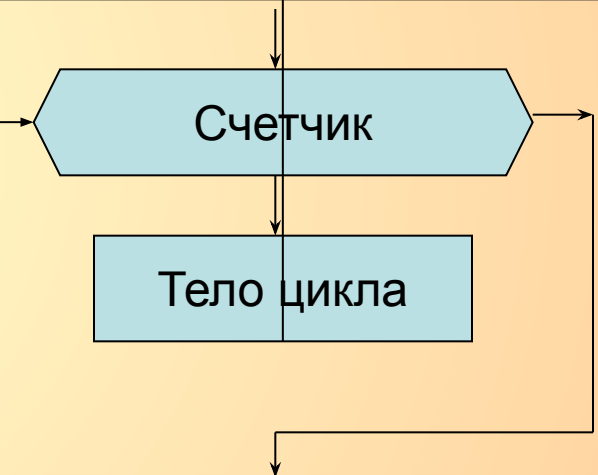
# Цикл со счетчиком.

Когда заранее известно, какое число повторений тела цикла необходимо выполнить, можно воспользоваться циклической инструкцией (оператором цикла со счетчиком) For . . . Next.

## VISUAL BASIC

```
For Счетчик=НачЗнач To КонЗнач [Step шаг]  
Next [Счетчик]
```

## Блок-схема



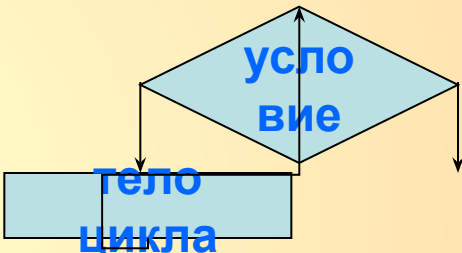
# Циклы с условием.

Часто бывает так, что необходимо повторить тело цикла, но заранее неизвестно, какое количество раз это надо сделать. В таких случаях количество повторений зависит от некоторого условия. Такой цикл реализуется с помощью инструкции **Do ... Loop.**

Условие выхода из цикла можно поставить в начале, перед телом цикла. Такой цикл называется *циклом с предусловием*

Условие выхода из цикла можно поставить в конце, после тела цикла. Такой цикл называется *циклом с постусловием*

# Циклы с предусловием.

| VISUAL BASIC                                                                                                                                                        | Блок-схема                                                                          |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <p>Do While    <b>Условие</b><br/>              <b>Тело цикла</b></p> <p>Loop</p> <p>Do Until    <b>Условие</b><br/>              <b>Тело цикла</b></p> <p>Loop</p> |  |

# Циклы с постусловием.

## VISUAL BASIC

Do

**Тело цикла**

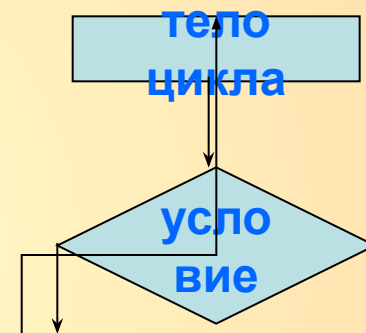
Loop While **Условие**

Do

**Тело цикла**

Loop Until **Условие**

## Блок-схема



# Повторим

- Что такое алгоритм?
- Перечислите виды алгоритмов.
- Отметьте основные способы описания алгоритмов:
  - а) блок-схема;
  - б) словесный;
  - в) с помощью нормальных форм;
  - г) с помощью граф-схем.

А) блок-схема;

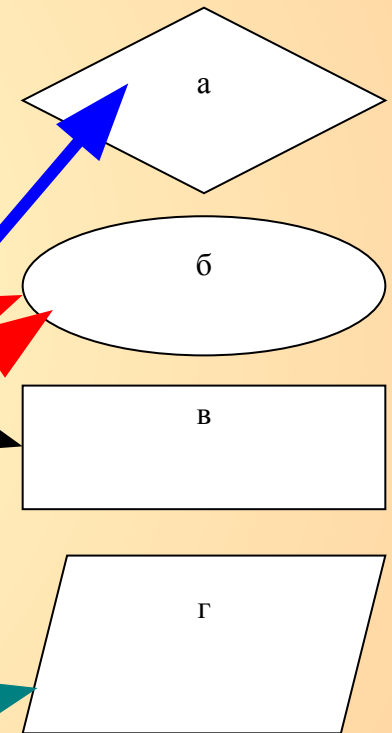
б) словесный;

- В какой алгоритмической структуре та или иная серия команд выполняется в зависимости от истинности условия?
- .Какой тип алгоритмической структуры необходимо применить, если последовательность команд выполняется или не выполняется?
- а) ветвление;
- б) цикл;
- в) выбор;
- г) линейный алгоритм.

- а) ветвление;
- б) цикл;
- в) выбор;

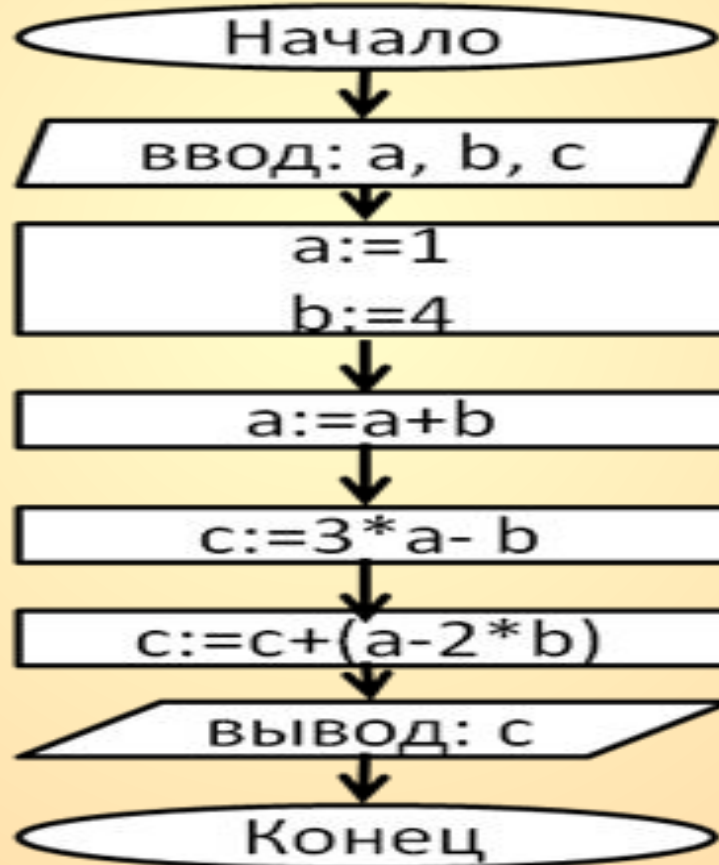
а

- Соотнесите высказывание номеру ответа:
- 1) выполнение действий
- 2) конец программы
- 3) начало программы
- 4) проверка условий
- 5) вывод данных



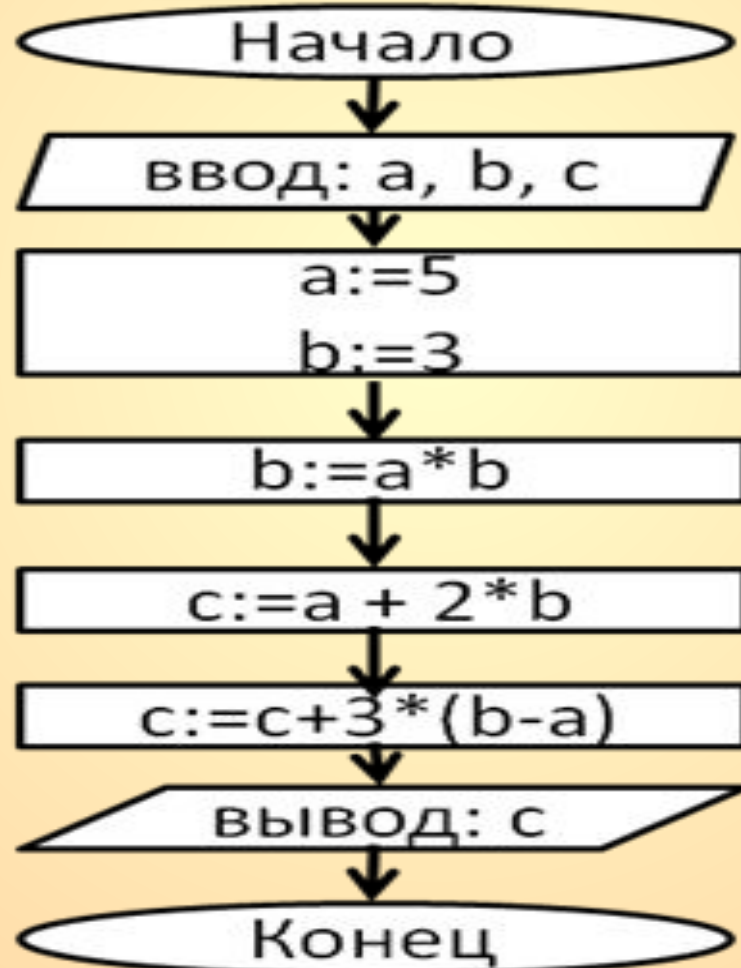
# Найти значение $c$ в ходе выполнения алгоритма

## 1 вариант

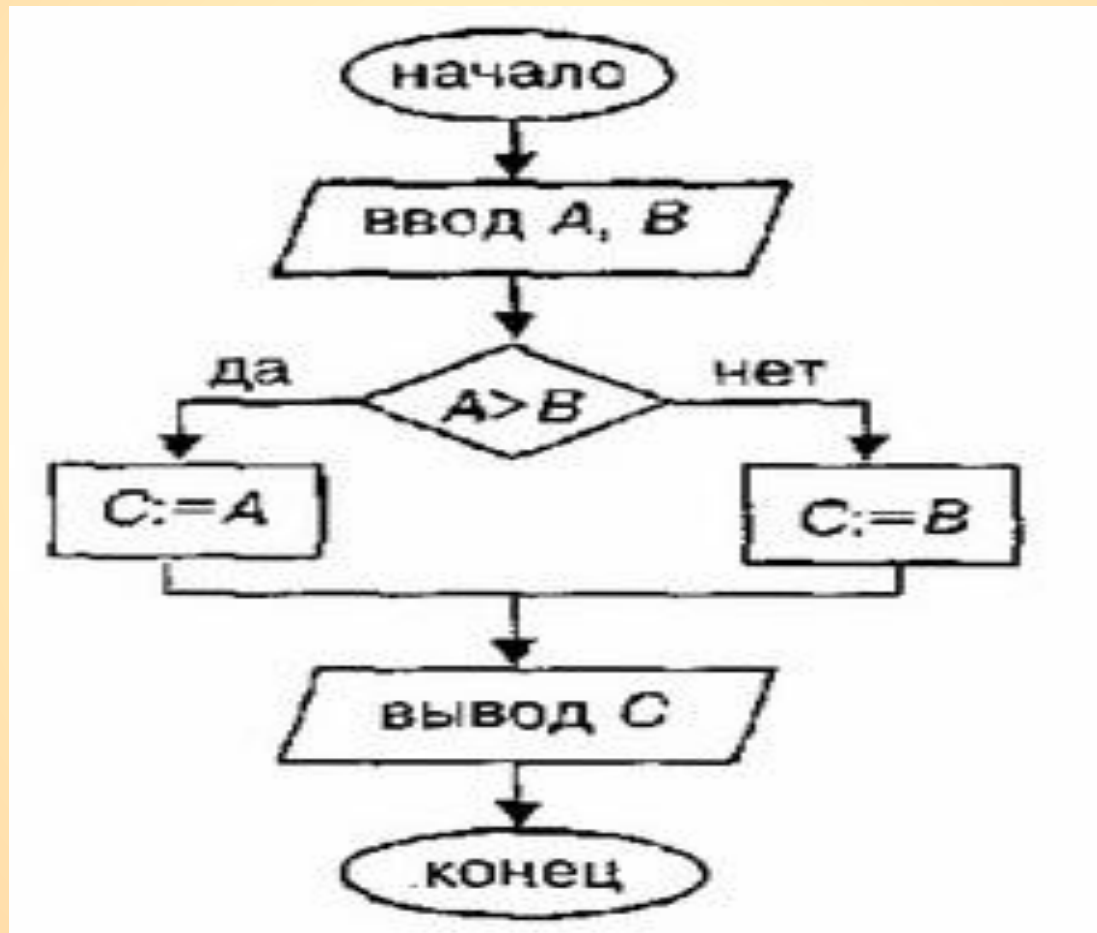


# Найти значение $c$ в ходе выполнения алгоритма

## 2 вариант

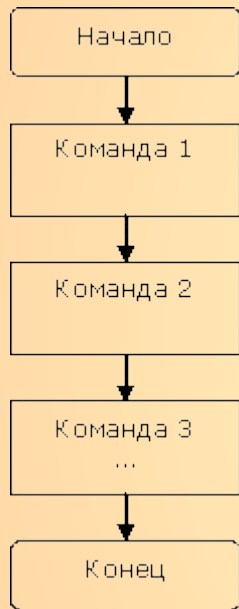


Даны два числа; выбрать большее из них.  $A = 5$ ,  $B = 8$ , вывести  $C$ .

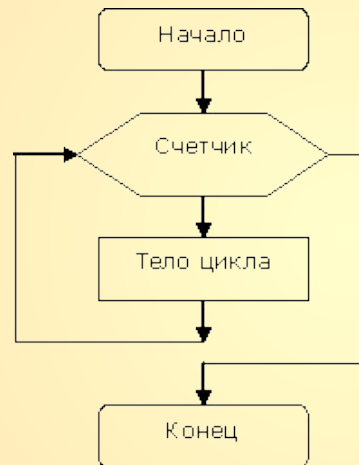


# . Какая из блок-схем является циклом?

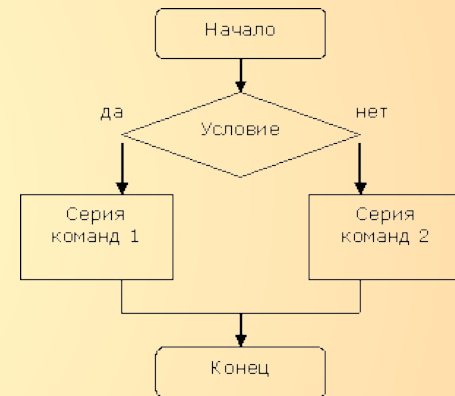
1.



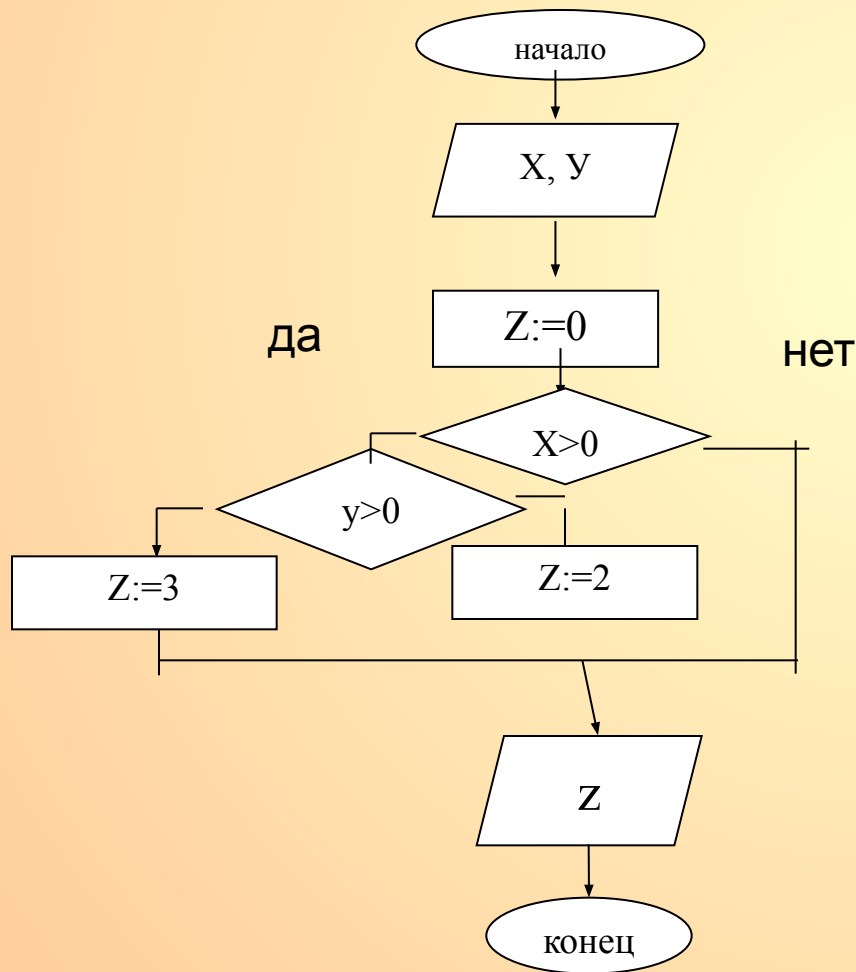
2



3



Какое значение получит переменная Z в результате выполнения следующего алгоритма?



А)  $x=1, y=1$

Б)  $x=1, y=-1$

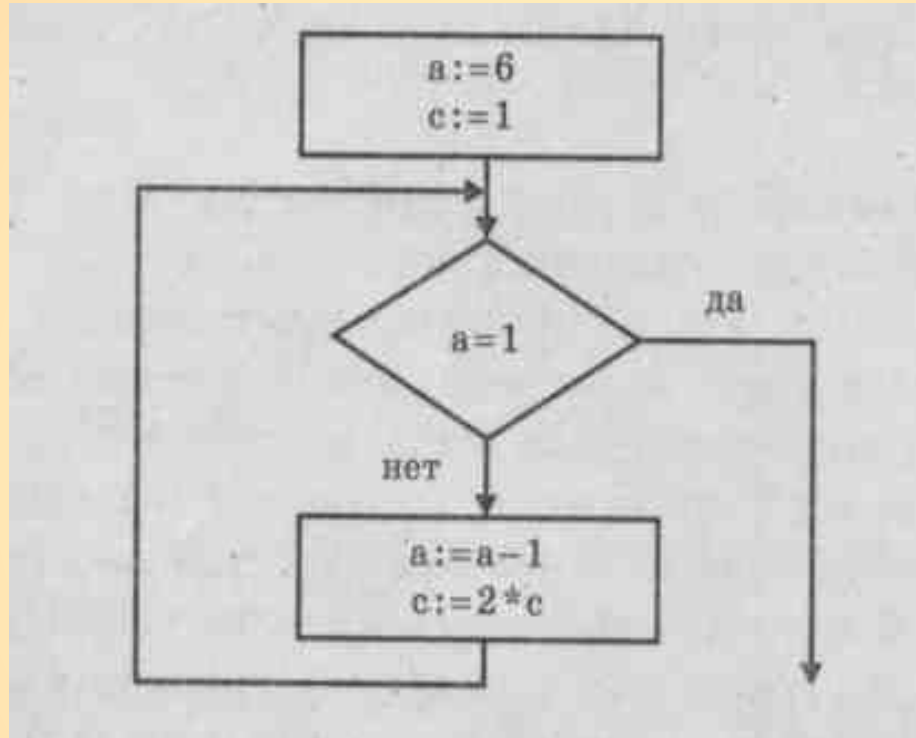
В)  $x=-1, y=1$

# Ответ

- A)  $Z:=3$
- Б)  $Z:=2$
- В)  $Z:=0$

# Определите значение переменной $c$ после выполнения фрагмента алгоритма

- :

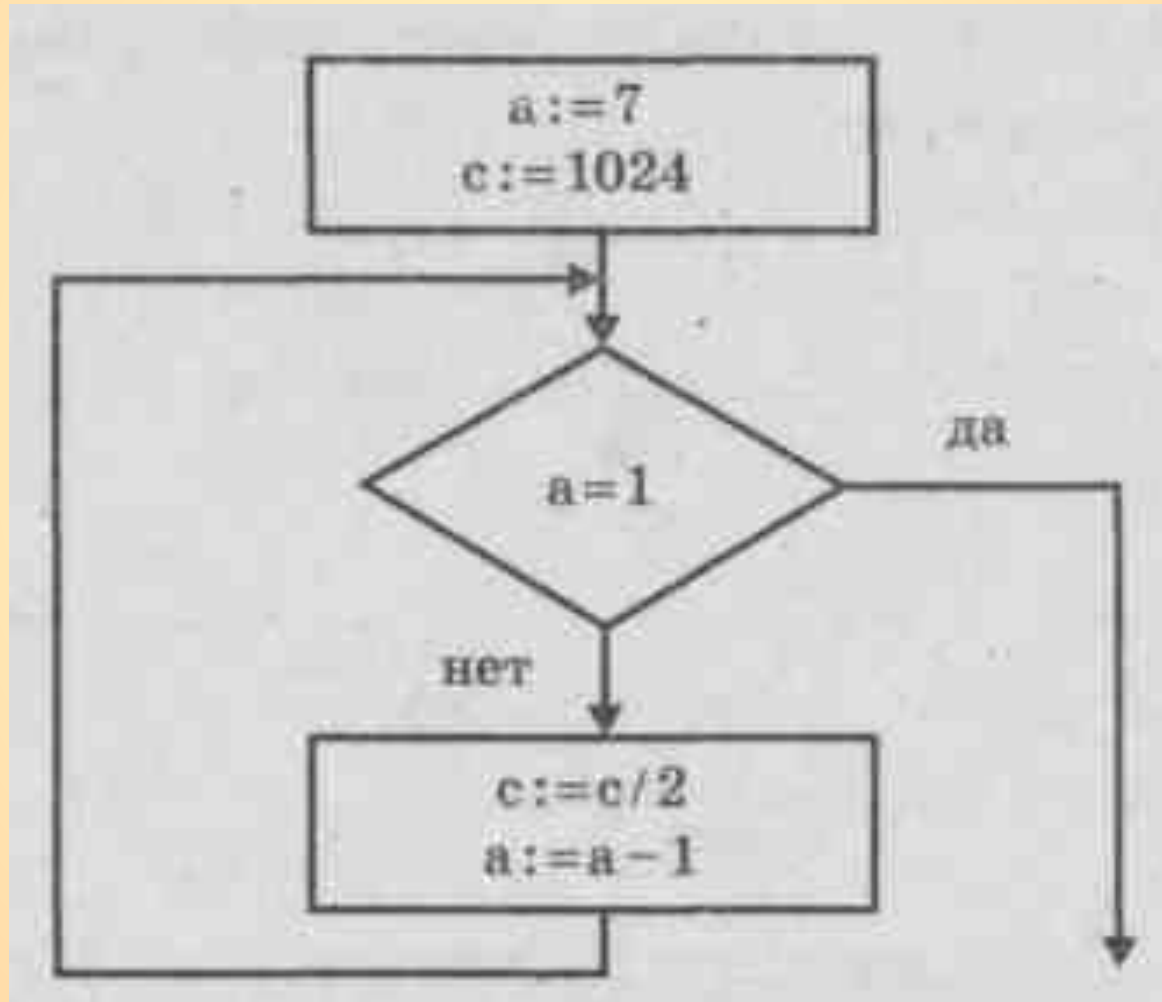


- Примечание:* знаком  $:=$  обозначена операция присваивания, знаком  $*$  обозначена операция умножения.

# Выполнение алгоритма

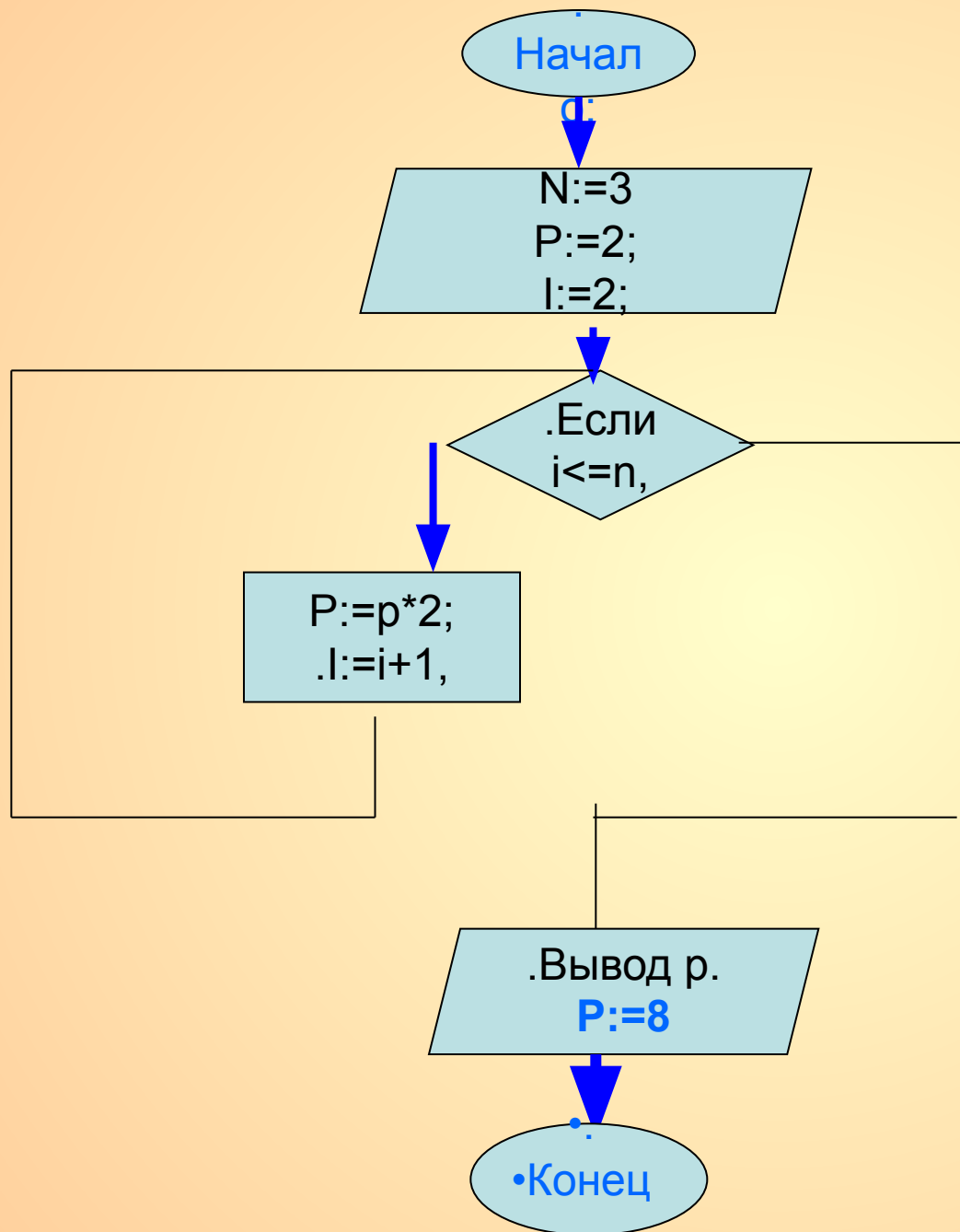
1.  $6=1$  (нет), значит  $a:=6-1$ ; то есть  $a:=5$   
 $c:=2*1$ , то есть  $c:=2$
2.  $5=1$  (нет), значит  $a:=5-1$ , то есть  $a:=4$   
 $c:=2*2$ , то есть  $c:=4$
3.  **$4=1$ (нет)** значит  $a:=4-1$ , то есть  $a:=3$   
 $c:=2*4$ , то есть  $c:=8$
4.  **$3=1$ (нет)** значит  $a:=3-1$ , то есть  $a:=2$   
 $c:=2*8$ , то есть  $c:=16$
5.  **$2=1$ (нет)** значит  $a:=2-1$ , то есть  $a:=1$   
 $c:=2*16$ , то есть  $c:=32$
6.  **$1=1$ (да)** значит  $c:=32$

Определите значение переменной  $c$  после выполнения фрагмента алгоритма

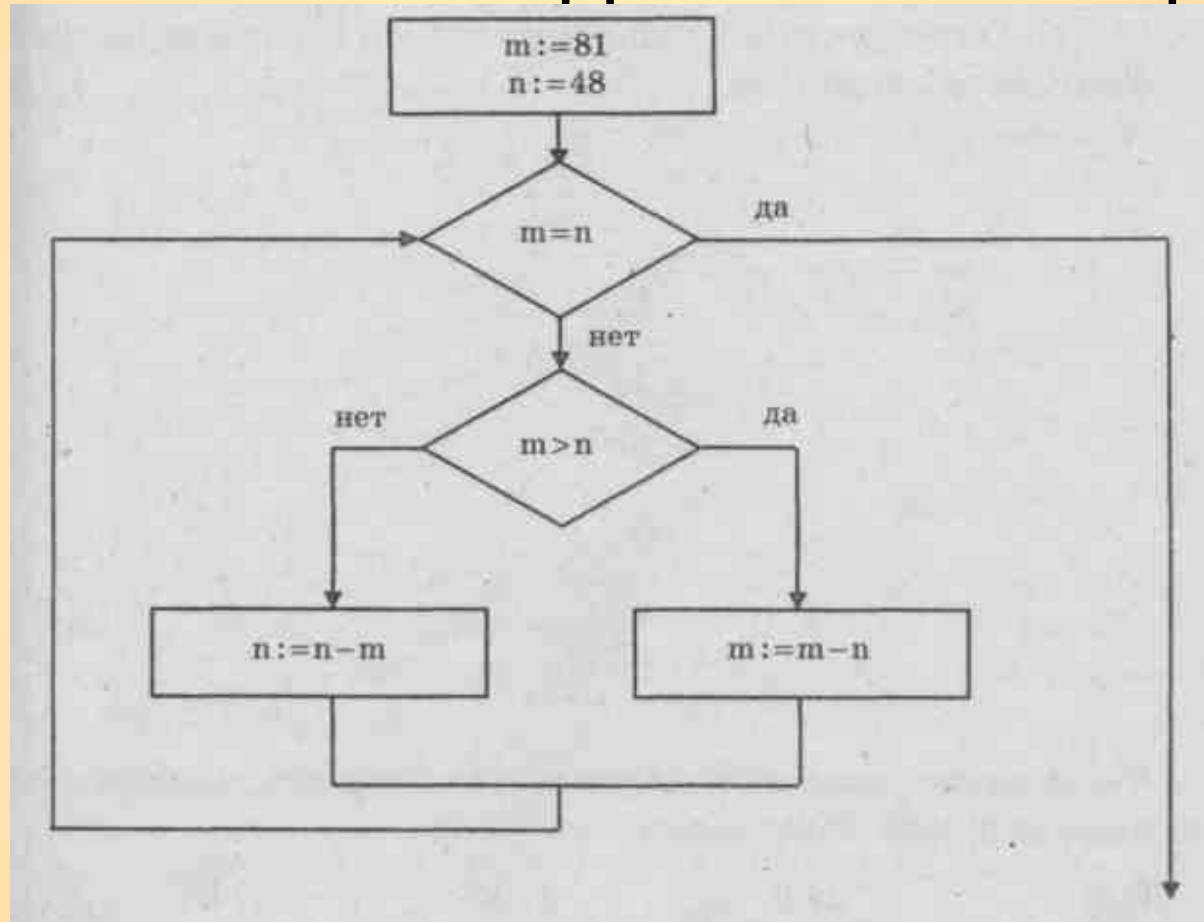


По словесному алгоритму вычисления  $2^n$  составьте блок-схему алгоритма.

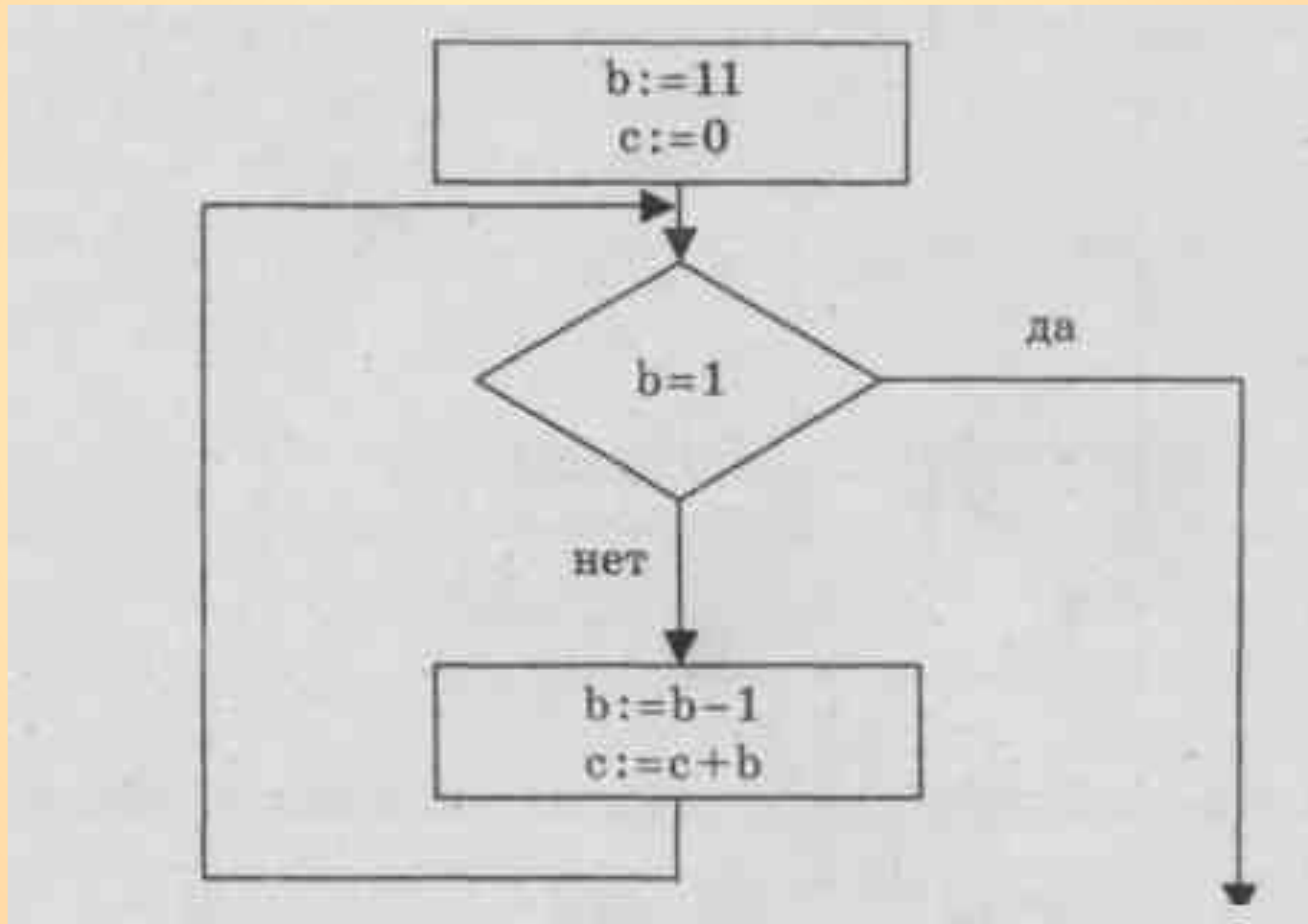
- 1.Начало;
- 2.Задать  $n$ ;
- 3. $P:=2$ ;
- 4. $i:=2$ ;
- 5.Если  $i \leq n$ , то п.6, иначе п.8;
- 6. $P:=P*2$ ;
- 7. $i:=i+1$ , перейти к п.5.;
- 8.Вывод  $P$ .
- 9.Конец



Определите значение переменной  $t$  после выполнения фрагмента алгоритма



Определите значение переменной  $c$  после выполнения фрагмента алгоритма



Определите значение переменной  $c$  после выполнения фрагмента алгоритма:

