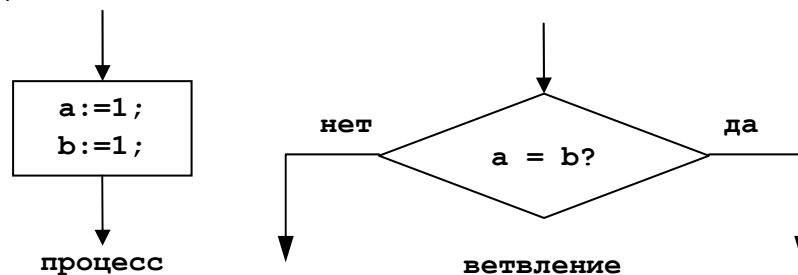


В2 (базовый уровень, время – 3 мин)

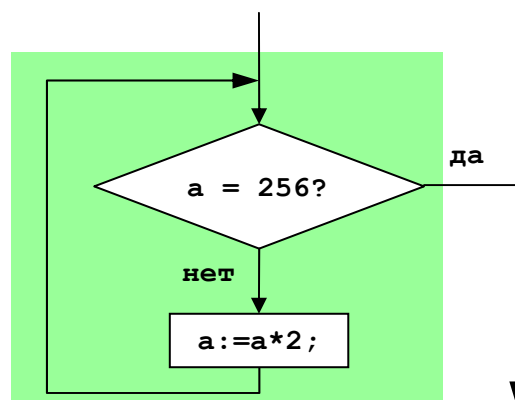
Тема: Блок-схемы алгоритмов. Переменные, присваивание значений. Ветвления. Организация циклов с помощью блока «ветвление».

Что нужно знать:

- переменная – это величина, которая имеет имя, тип и значение; переменная может изменяться во время выполнения программы
- оператор присваивания (в Паскале обозначается сочетанием символов « := ») служит для записи нового значения в переменную (для изменения ее значения)
- если в переменную записывают новое значение, старое стирается
- знаки +, -, *, / используются для обозначения операций сложения, вычитания, умножения и деления
- запись вида $a := a + 2$; – это не уравнение, а команда «прочитать текущее значение переменной a , добавить к нему 2 и записать результат обратно в переменную a »;
- для наглядной записи небольших алгоритмов используют блок-схемы; они состоят из блоков разного назначения и соединительных линий со стрелками, которые показывают порядок выполнения блоков
- в задачах ЕГЭ встречаются два блока: *процесс* (выполнение некоторых действий) и *ветвление* (условие, в зависимости от которого выполнение алгоритма продолжается по одной или другой «ветке»)

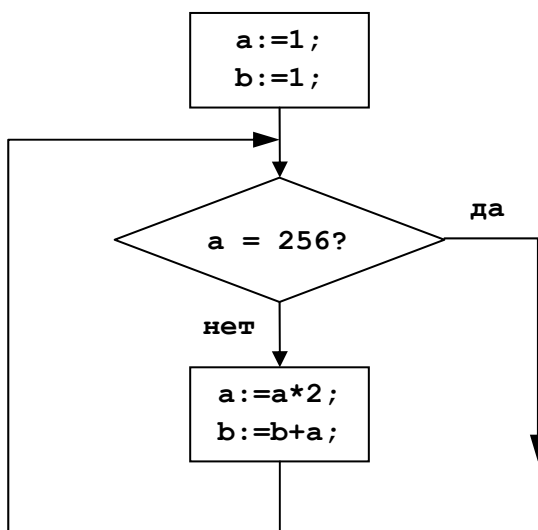


- с помощью ветвления можно организовать *цикл* (многократное выполнение одинаковых действий), в этом случае в блок-схеме будет соединительная линия, идущая «в обратном направлении» (петля, замкнутый контур)
- цикл на рисунке (выделен зеленым фоном) закончится только тогда, когда выполнится условие $a = 256$



Пример задания:

Запишите значение переменной **b** после выполнения фрагмента алгоритма:



Решение (вариант 1, ручная прокрутка):

- 1) по схеме видим, что алгоритм содержит цикл (есть петля, контур)
- 2) ручную прокрутку удобнее всего выполнять в виде таблицы, в первом столбце будем записывать выполняемые команды, во втором и третьем – изменение значений переменных **a** и **b**

- 3) после выполнения первого блока получаем

	a	b
a:=1;	1	?
b:=1;		1

знак вопроса означает, что после выполнения первого оператора значение **b** не определено

- 4) затем выполняется проверка условия; поскольку **a** не равно 256, ответ на вопрос «**a = 256?**» будет «нет»:

	a	b
a:=1;	1	?
b:=1;		1
a = 256?	нет	

- 5) далее алгоритм уходит на выполнение тела цикла; здесь сначала меняется переменная **a**, а потом – **b**, причем нужно помнить, что для вычисления **b** используется новое значение **a**, равное 2, поэтому новое значение **b** равно $1 + 2 = 3$:

	a	b
a:=1;	1	?
b:=1;		1
a = 256?	нет	
a:=a*2;	2	
b:=b+a;		3

- 6) после этого по стрелке переходим на проверку условия; поскольку **a = 2**, ответ на вопрос «**a = 256?**» снова будет «нет», и выполняется очередной шаг цикла:

	a	b
a:=1;	1	?
b:=1;		1
a = 256?	нет	

$a := a * 2;$	2	
$b := b + a;$		3
$a = 256?$	нет	
$a := a * 2;$	4	
$b := b + a;$		7

- 7) аналогично можно выполнить вручную все шаги цикла, результаты последнего из них выглядят так:

	a	b
$a := a * 2;$	256	
$b := b + a;$		511
$a = 256?$	да	

как только значение **a** стало равно 256, цикл завершает работу

- 8) таким образом, верный ответ – **511**.

Возможные проблемы:

- таблица получается длинной, много вычислений, можно запутаться
- нужно не забыть, что при выполнении двух операторов в теле цикла к значению **b** добавляется уже новое значение **a**, полученное в предыдущей строке
- не перепутайте переменную, значение которой нужно определить (можно по ошибке вписать в ответ полученное значение **a**)

Решение (вариант 2, анализ алгоритма):

- 1) «прокрутив» начало алгоритма, можно заметить, что последовательные значения **a** – это степени двойки
 $a = 1, 2, 4, 8, \dots, 256$
- 2) поскольку оператор $b := b + a$ означает «взять текущее значение **b**, прибавить к нему текущее значение **a** и результат записать обратно в **b**», изменение **b** сводится к тому, что эти степени двойки складываются:
 $b = 1 + 2 + 4 + 8 + \dots + 256$
- 3) теперь можно, конечно, сложить эти числа вручную (их всего 9), но можно заметить (или вспомнить), что сумма всех последовательных степеней двойки, начиная с 1, на единицу меньше, чем следующая степень двойки¹ (первая, не вошедшая в сумму, здесь – 512); это легко проверяется по начальной части таблицы
- 4) таким образом, верный ответ $512 - 1 = 511$.

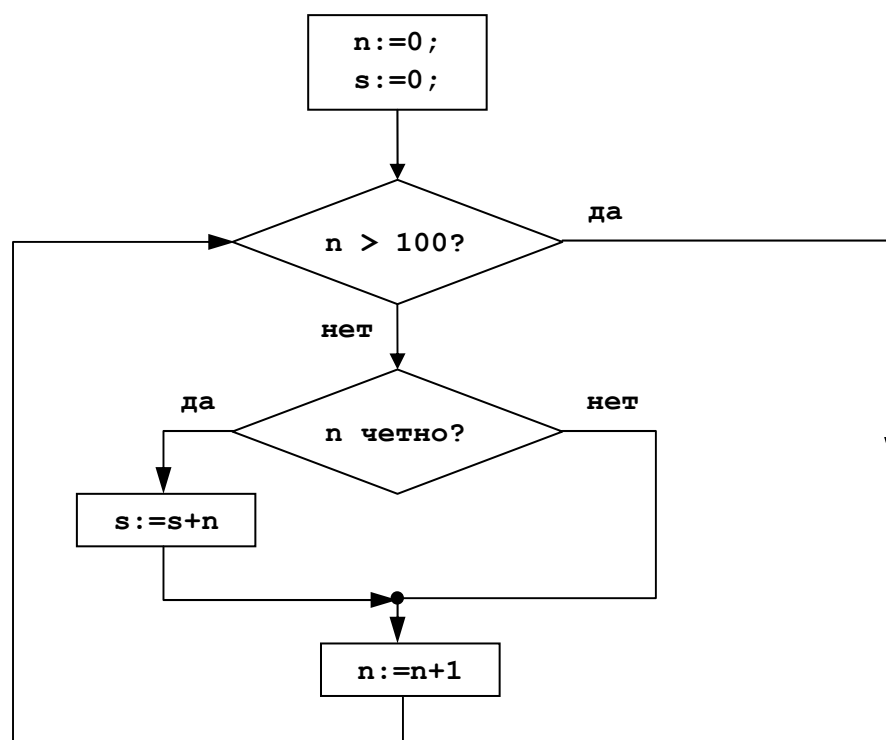
Возможные проблемы:

- для такого анализа требуется некоторое напряжение ума, здесь не обойтись формальным выполнением каких-то заученных действий
- не всегда удастся найти короткое решение, «свернув» алгоритм таким образом (в этом случае поможет ручная прокрутка)

Еще пример задания:

*Запишите значение переменной **s** после выполнения фрагмента алгоритма:*

¹ Попробуйте доказать это, используя знания по теме «Двоичная система счисления».



Решение (анализ алгоритма):

- 1) «прокрутив» начало алгоритма, можно заметить, что в переменной s суммируются все четные числа от 0 до 100 включительно:

$$s = 0 + 2 + 4 + \dots + 98 + 100$$
- 2) теперь можно подсчитать эту сумму вручную, но можно легко ошибиться в расчетах
- 3) попробуем сгруппировать числа по парам²:
 0 и 100, 2 и 98, 4 и 96, ..., 48 и 52
 без пары остается только число 50
- 4) заметим, что сумма двух чисел в каждой паре равна 100
- 5) очевидно, что таких пар – 25 штук (первое число изменяется от 0 до 48 с шагом 2, это все равно, что от 0 до 24 с шагом 1)
- 6) теперь остается перемножить 25 на 100 и добавить 50
- 5) таким образом, верный ответ $100 \cdot 25 + 50 = 2550$.
- 6) заметим, что можно было составить пары иначе:
 2 и 100, 4 и 98, 6 и 96, ..., 50 и 52
 сумма чисел в каждой паре – 102, всего 25 пар, поэтому сумма равна $102 \cdot 25 = 2550$.

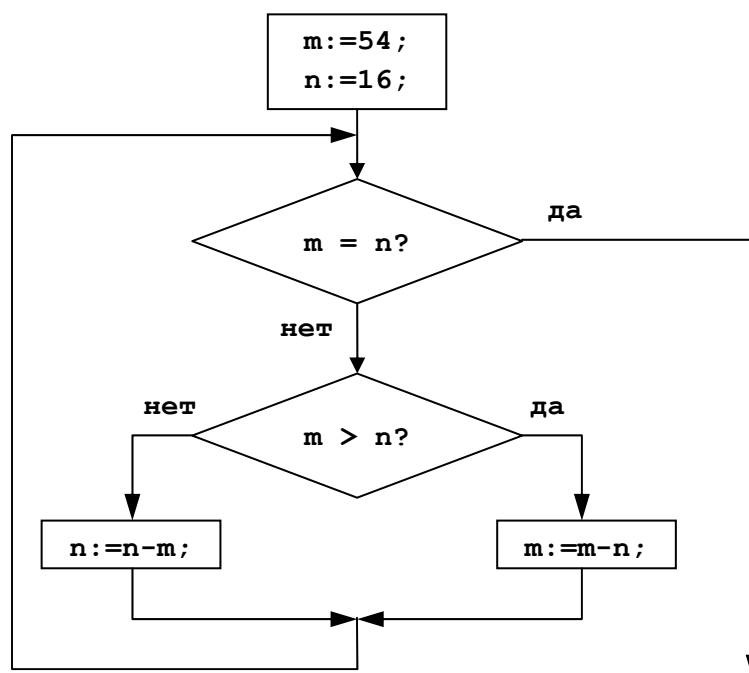
Возможные проблемы:

- если использовать ручную прокрутку при большом количестве шагов цикла, велик шанс ошибиться в арифметике; в этом случае нужно постараться найти способ «свернуть» вычисления, свести их к формуле или как-то сгруппировать слагаемые

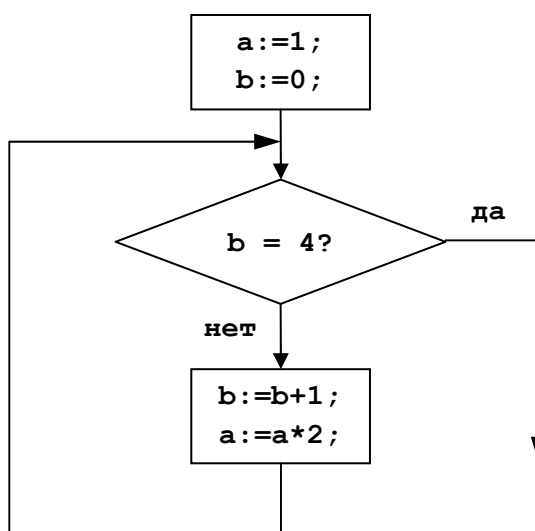
² Согласно легенде, такой прием использовал, будучи школьником, великий немецкий математик К.Ф. Гаусс. Составители ЕГЭ считают, что вы так же талантливы.

Задачи для тренировки³:

1) Определите значение переменной **m** после выполнения фрагмента алгоритма.



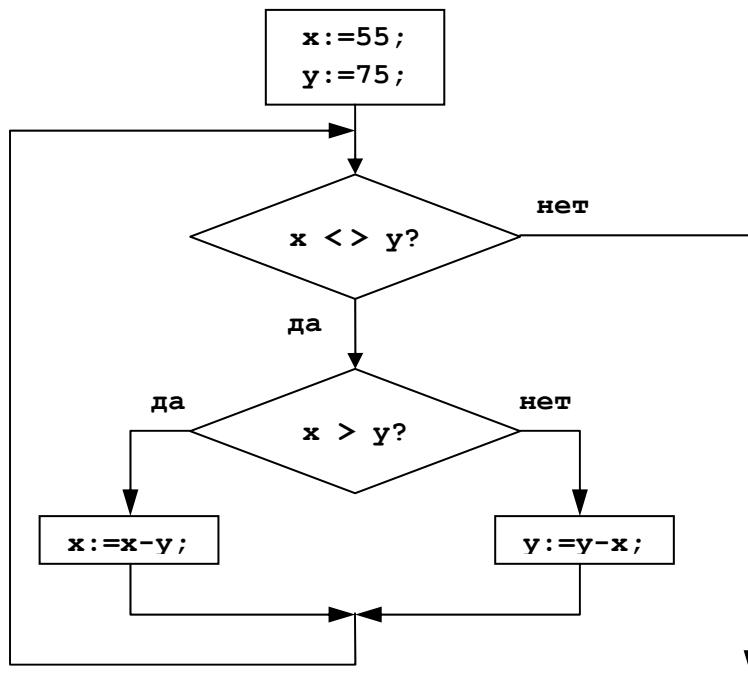
2) Определите значение переменной **a** после выполнения фрагмента алгоритма.



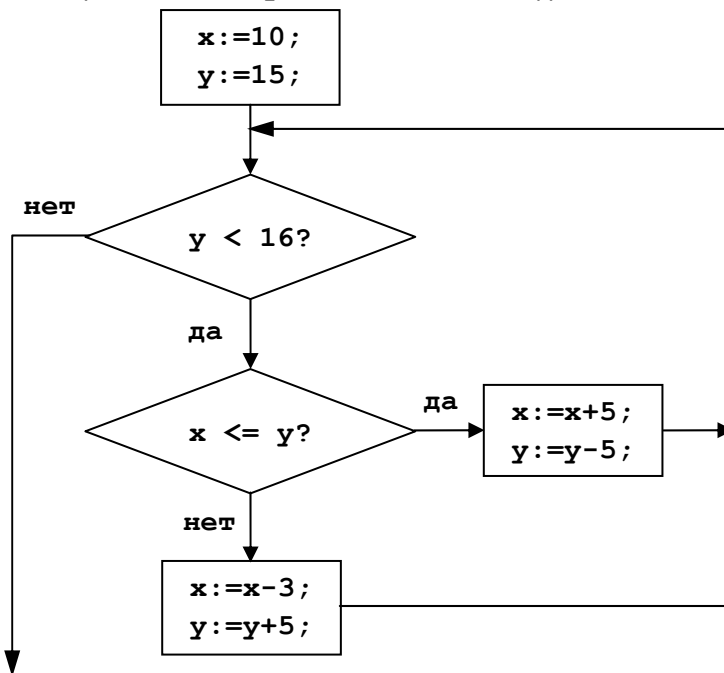
³ Источники заданий:

1. Демонстрационные варианты ЕГЭ 2004-2011 гг.
2. Гусева И.Ю. ЕГЭ. Информатика: раздаточный материал тренировочных тестов. — СПб: Тригон, 2009.
3. Крылов С.С., Лещинер В.Р., Якушкин П.А. ЕГЭ-2010. Информатика. Универсальные материалы для подготовки учащихся / под ред. В.Р. Лещинера / ФИПИ. — М.: Интеллект-центр, 2010.
4. Якушкин П.А., Ушаков Д.М. Самое полное издание типовых вариантов реальных заданий ЕГЭ 2010. Информатика. — М.: Астрель, 2009.
5. Самылкина Н.Н., Островская Е.М. ЕГЭ 2011. Информатика. Тематические тренировочные задания. — М.: Эксмо, 2010.
6. Якушкин П.А., Лещинер В.Р., Кириенко Д.П. ЕГЭ 2011. Информатика. Типовые тестовые задания. — М.: Экзамен, 2011.

3) Определите значение переменной **х** после выполнения фрагмента алгоритма.



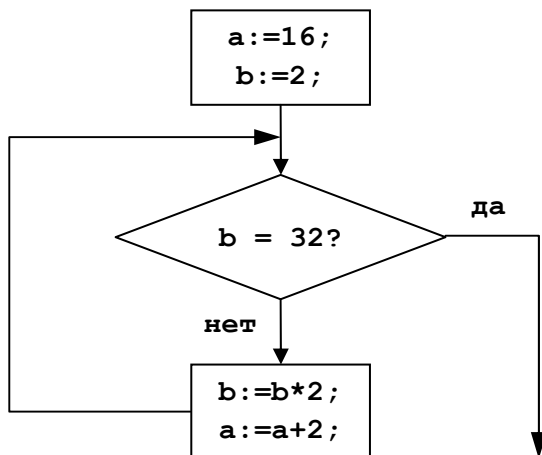
4) Определите значения переменных **х** и **у** после выполнения фрагмента алгоритма.



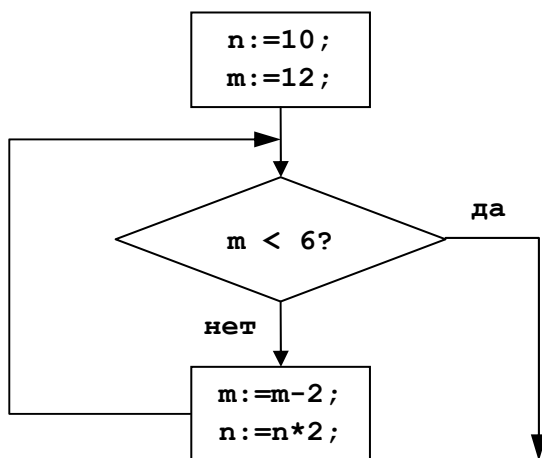
В ответ запишите номер правильного варианта:

- 1) $x=15, y=16$ 2) $x=20, y=13$ 3) $x=16, y=15$ 4) $x=13, y=20$

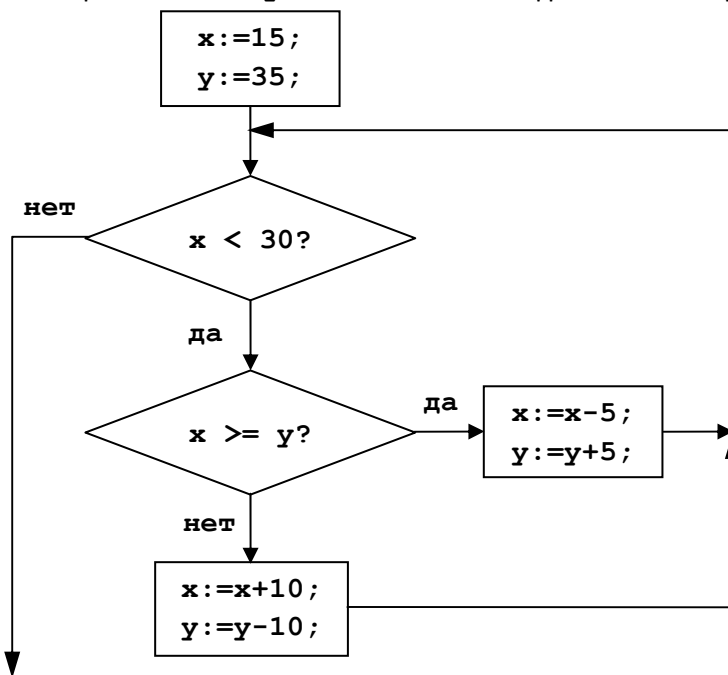
5) Определите значение переменной **a** после выполнения фрагмента алгоритма.



6) Определите значение переменной **n** после выполнения фрагмента алгоритма.



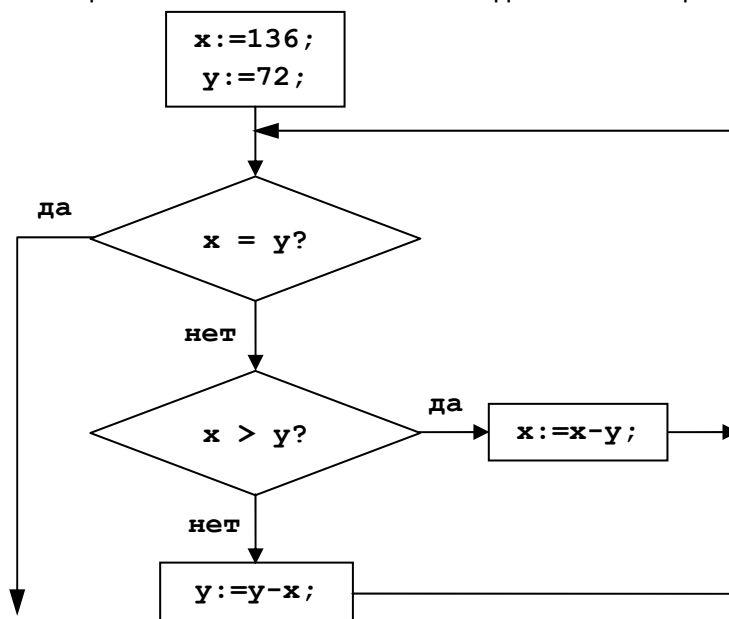
7) Определите значения переменных **x** и **y** после выполнения фрагмента алгоритма.



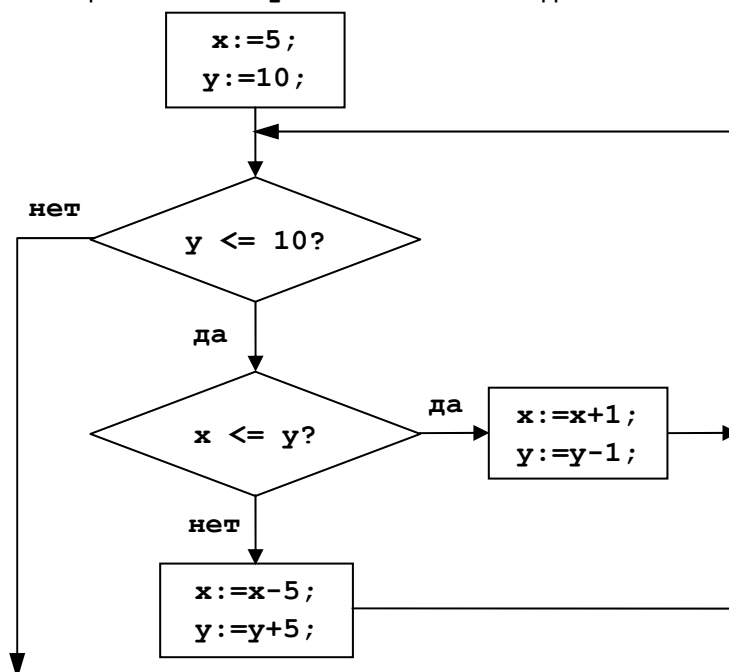
В ответ запишите номер правильного варианта:

- 1) x=25, y=25 2) x=20, y=30 3) x=30, y=20 4) x=30, y=30

8) Определите значение переменной **x** после выполнения фрагмента алгоритма.



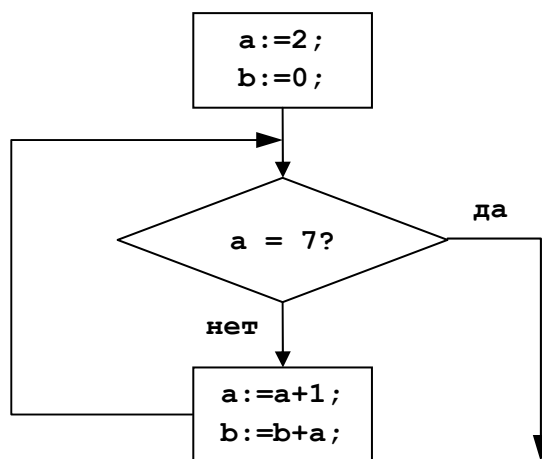
9) Определите значения переменных **x** и **y** после выполнения фрагмента алгоритма.



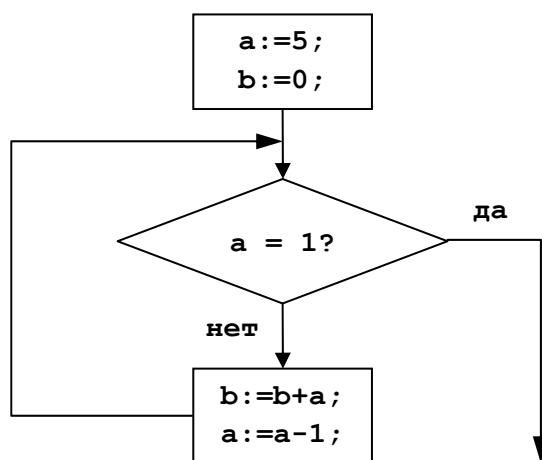
В ответ запишите номер правильного варианта:

- 1) x=5, y=15 2) x=3, y=12 3) x=10, y=5 4) x=9, y=15

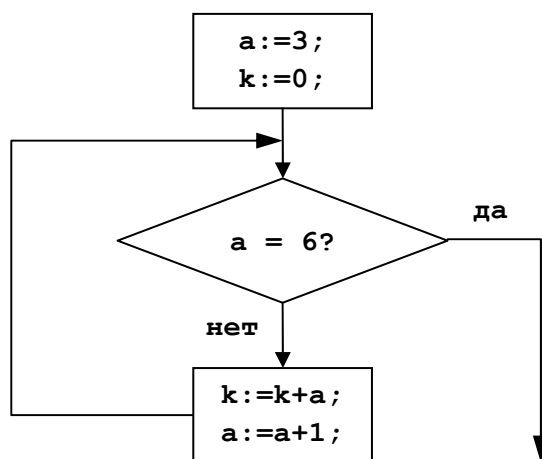
10) Определите значение переменной **b** после выполнения фрагмента алгоритма.



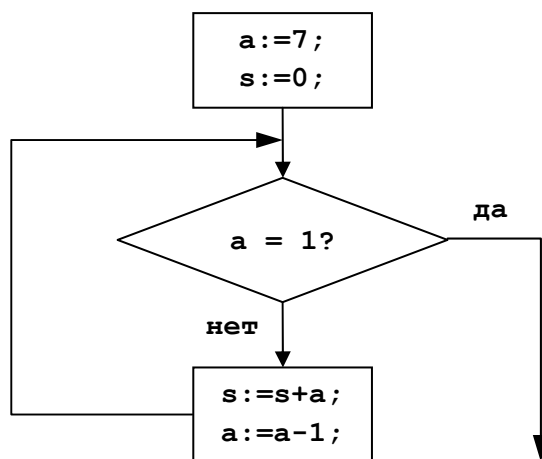
11) Определите значение переменной **b** после выполнения фрагмента алгоритма.



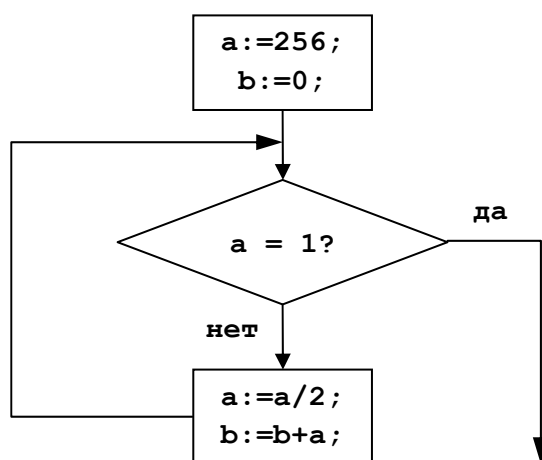
12) Определите значение переменной **k** после выполнения фрагмента алгоритма.



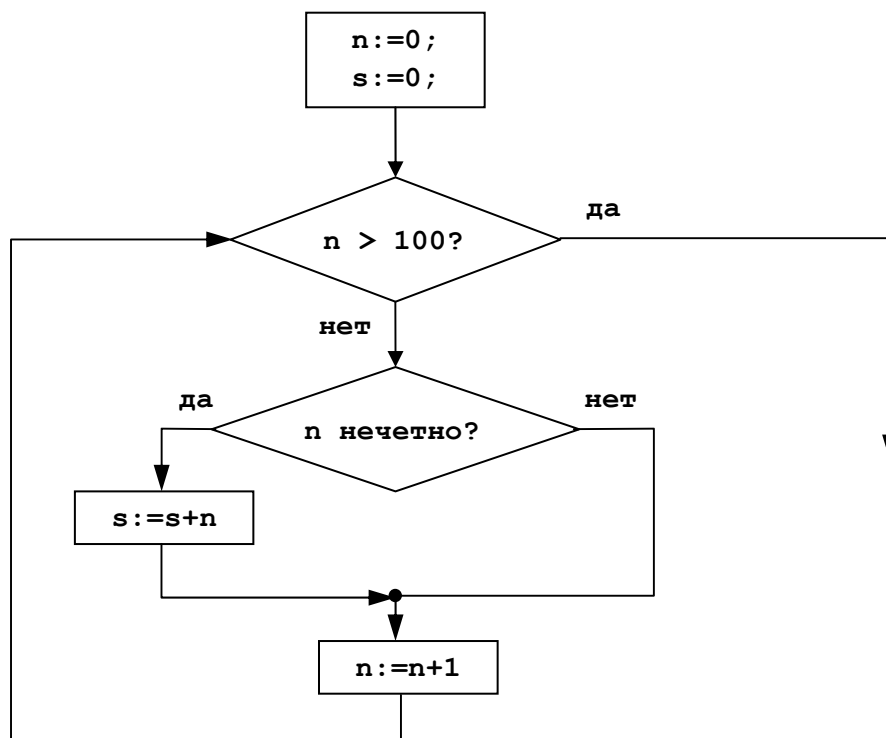
13) Определите значение переменной **s** после выполнения фрагмента алгоритма.



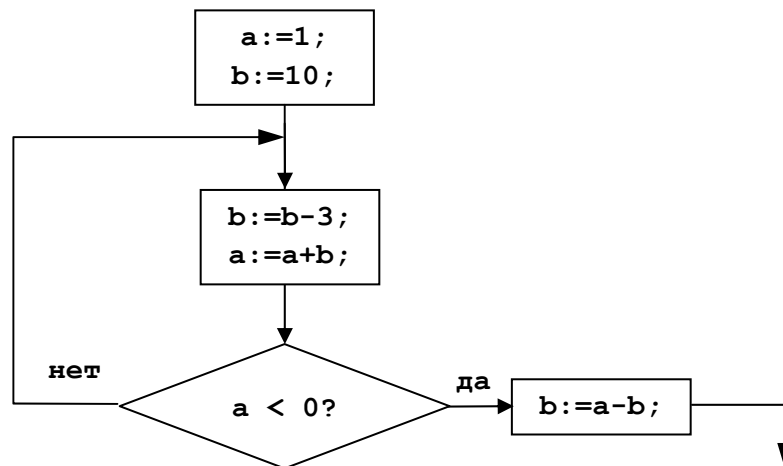
14) Определите значение переменной **b** после выполнения фрагмента алгоритма.



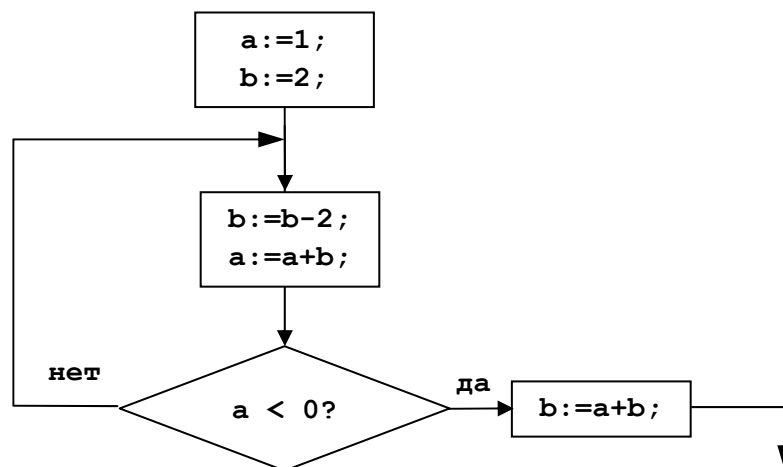
15) Запишите значение переменной **s** после выполнения фрагмента алгоритма:



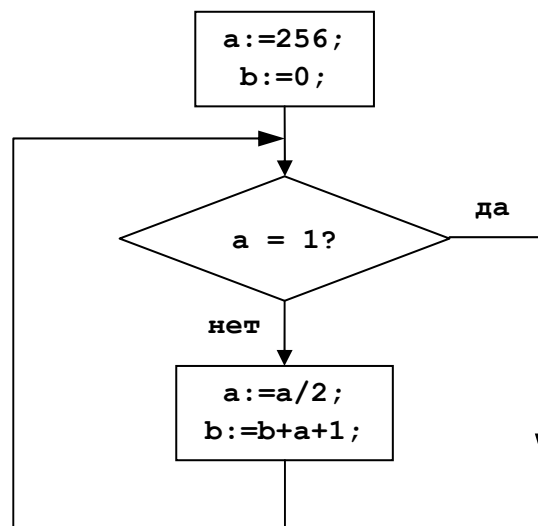
16) Определите значение переменной **b** после выполнения фрагмента алгоритма:



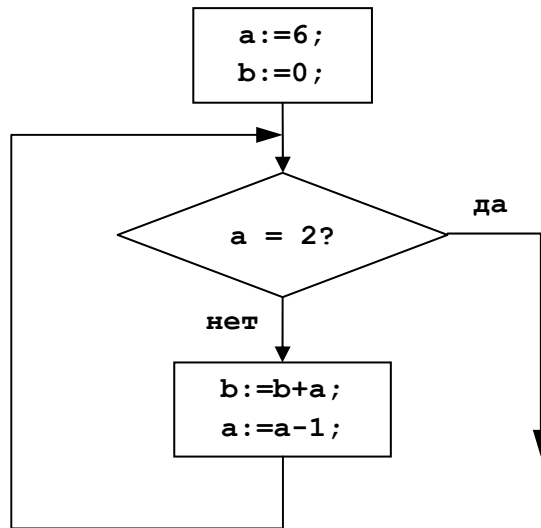
17) Определите значение переменной **b** после выполнения фрагмента алгоритма:



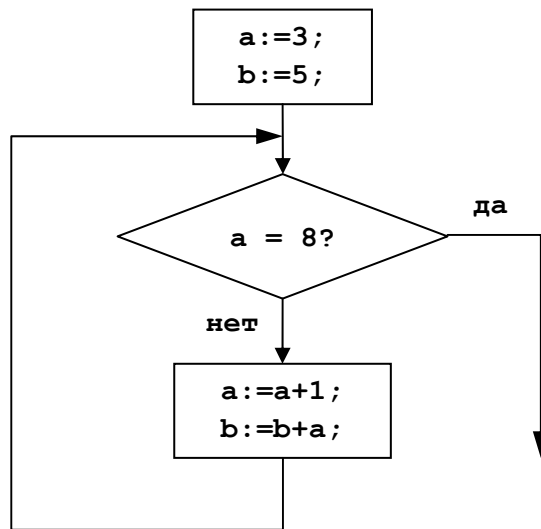
18) Определите значение переменной **b** после выполнения фрагмента алгоритма.



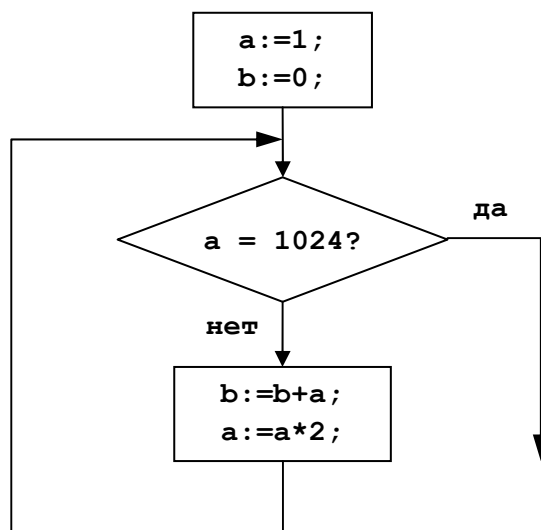
19) Определите значение переменной **b** после выполнения фрагмента алгоритма.



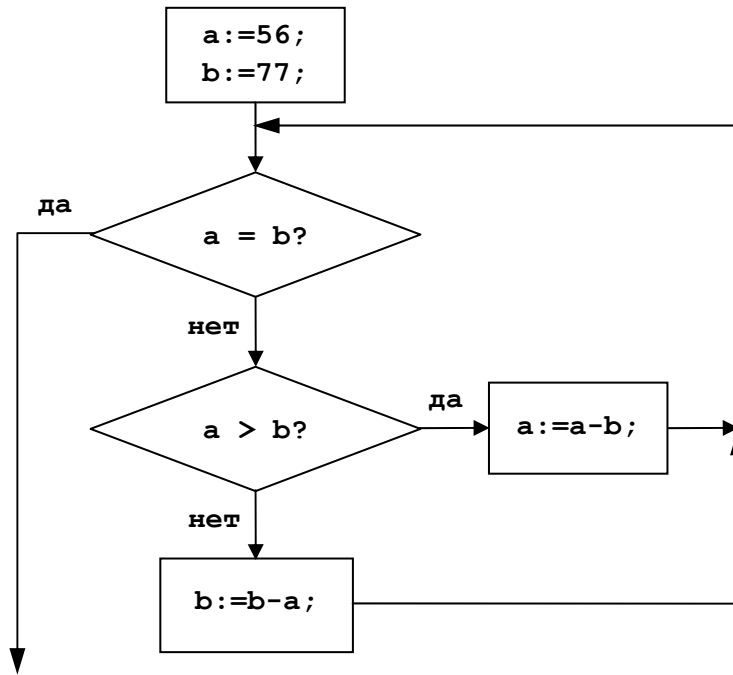
20) Определите значение переменной **b** после выполнения фрагмента алгоритма.



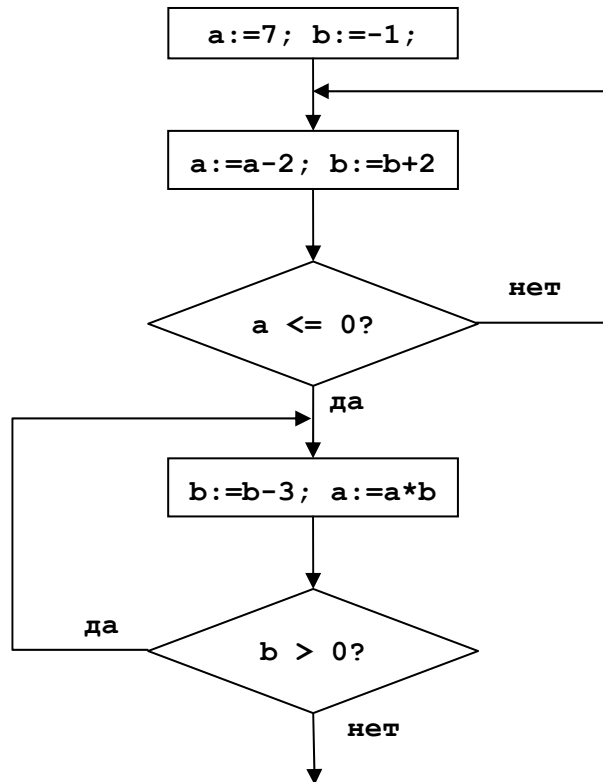
21) Определите значение переменной **b** после выполнения фрагмента алгоритма.



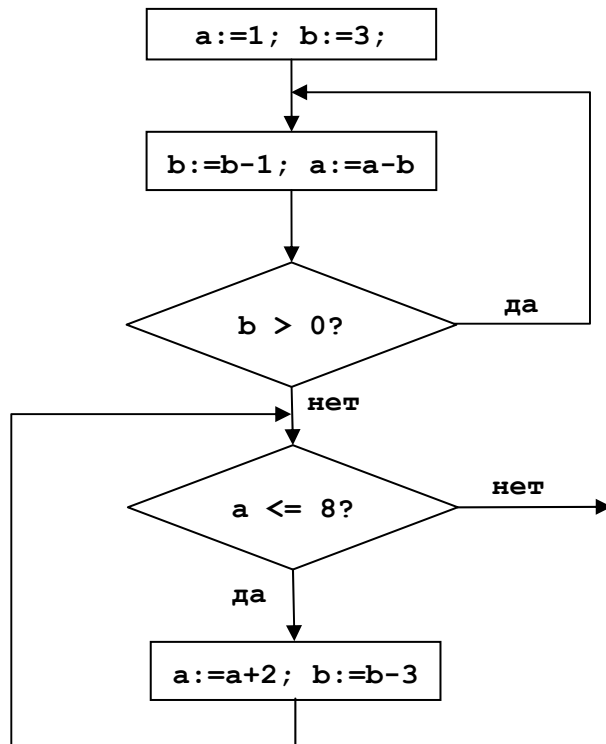
22) Определите значения переменной **a** после выполнения фрагмента алгоритма.



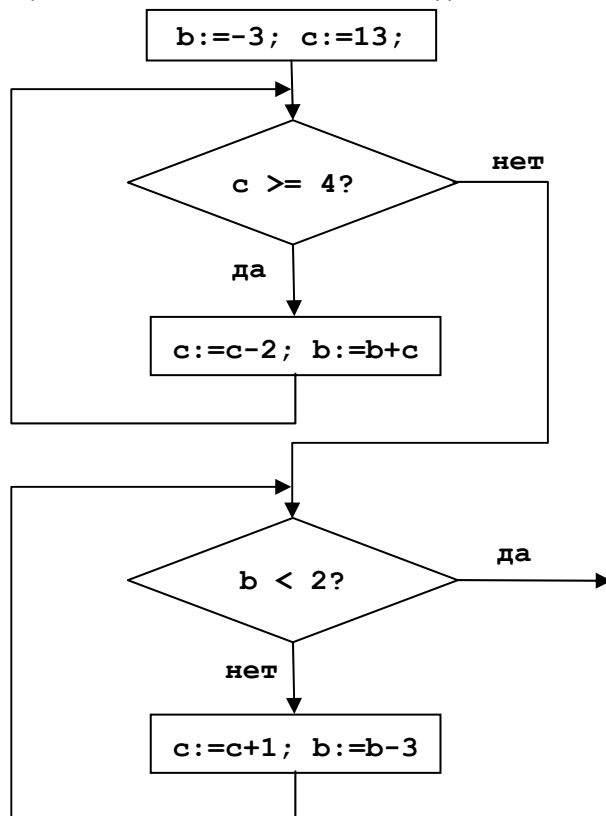
23) Определите значение переменной **a** после выполнения фрагмента алгоритма.



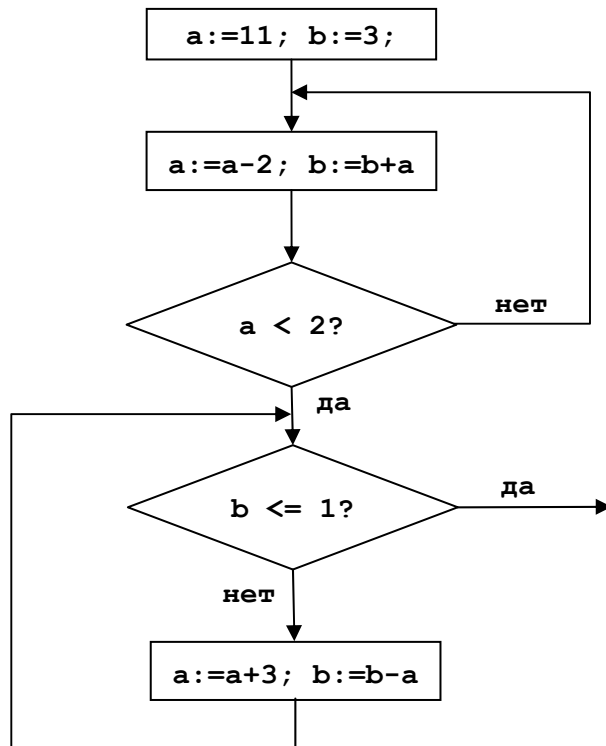
24) Определите значение переменной **b** после выполнения фрагмента алгоритма.



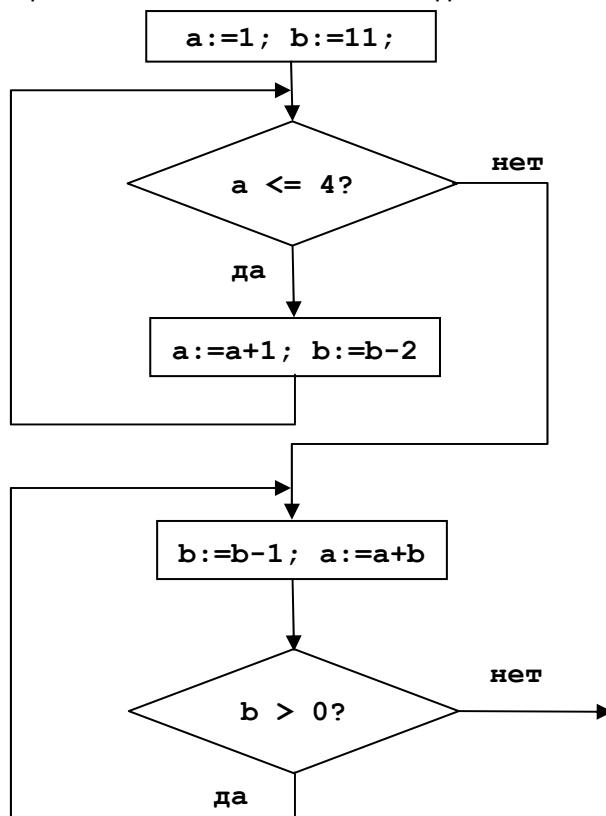
25) Определите значение переменной **c** после выполнения фрагмента алгоритма.



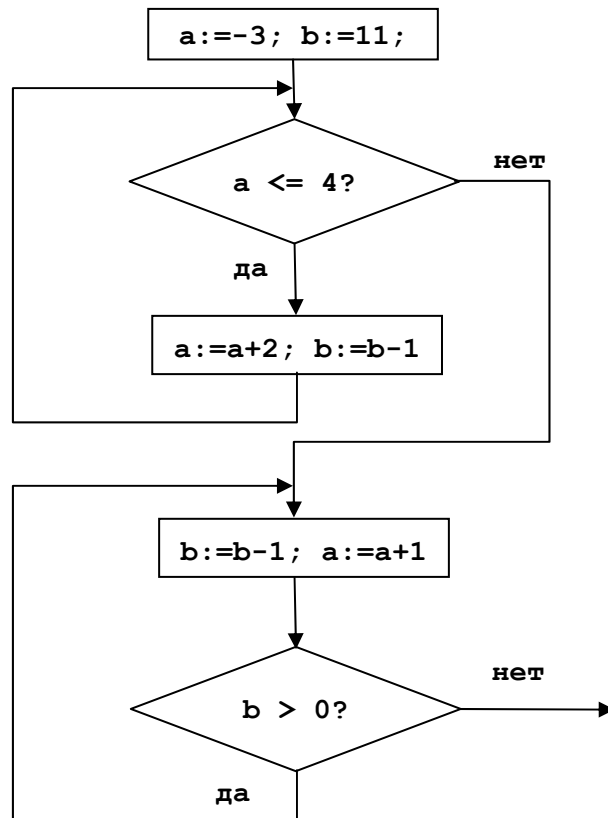
26) Определите значение переменной **b** после выполнения фрагмента алгоритма.



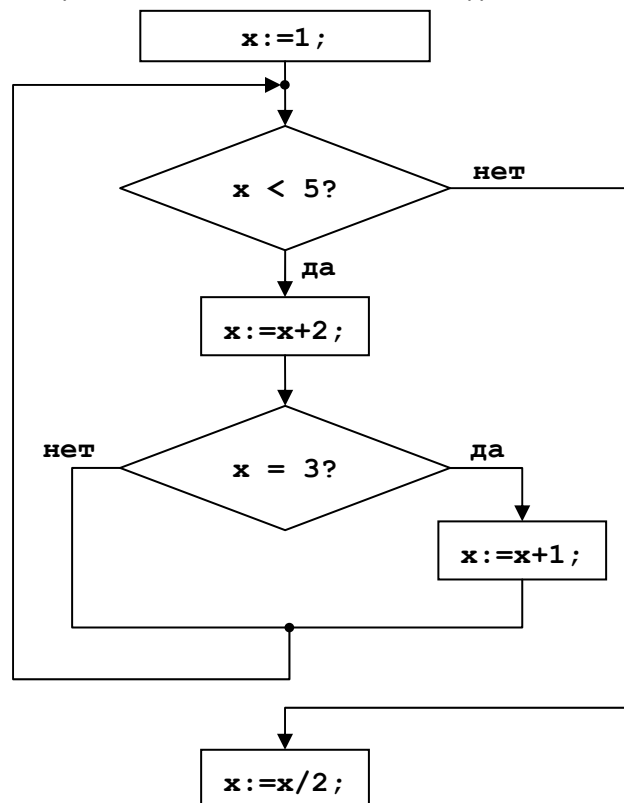
27) Определите значение переменной **a** после выполнения фрагмента алгоритма.



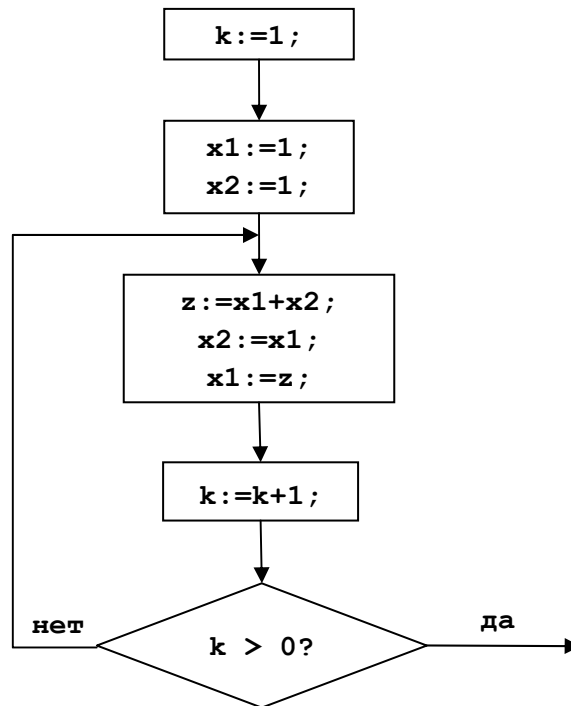
28) Определите значение переменной **a** после выполнения фрагмента алгоритма.



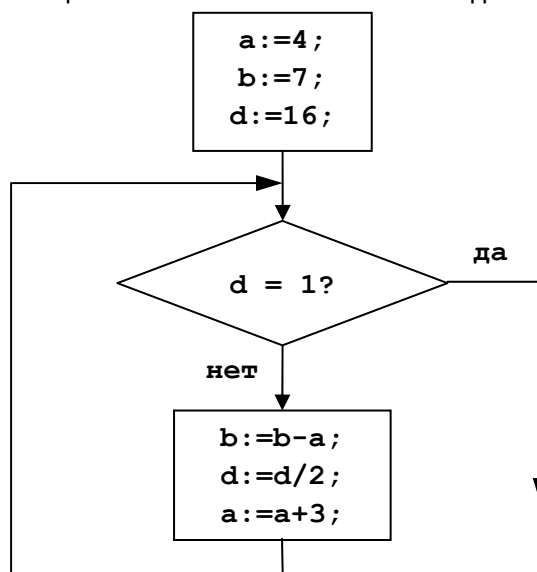
29) Определите значение переменной **x** после выполнения фрагмента алгоритма.



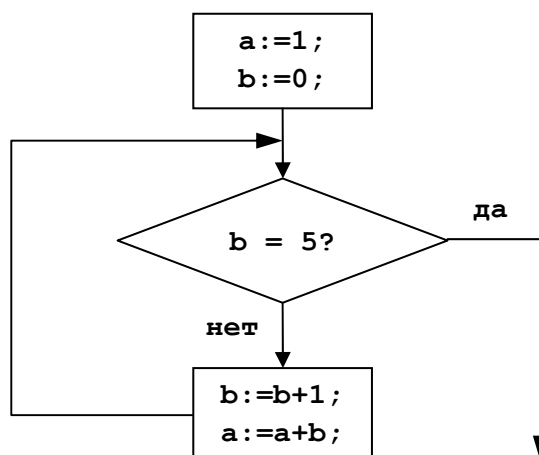
30) Определите значение переменной z после выполнения фрагмента алгоритма.



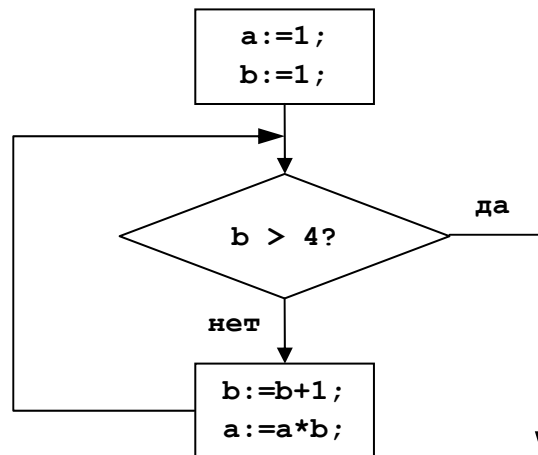
31) Определите значение переменной b после выполнения фрагмента алгоритма.



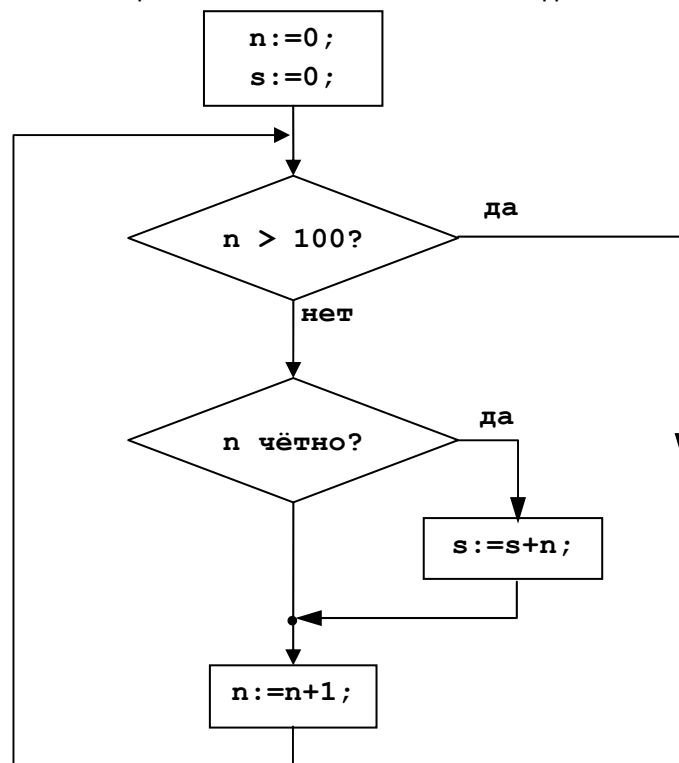
32) Определите значение переменной a после выполнения фрагмента алгоритма.



33) Определите значение переменной a после выполнения фрагмента алгоритма.



34) Определите значение переменной s после выполнения фрагмента алгоритма.



35) Определите значение переменной **C** после выполнения фрагмента алгоритма.

