

## A17 (повышенный уровень, время – 6 мин)

Тема: Работа с массивами и матрицами в языке программирования<sup>1</sup>.

Что нужно знать:

- работу цикла `for` (цикла с переменной)
- массив – это набор однотипных элементов, имеющих общее имя и расположенных в памяти рядом
- для обращения к элементу массива используют квадратные скобки, запись `A[i]` обозначает элемент массива `A` с номером (индексом) `i`
- матрица (двухмерный массив) – это прямоугольная таблица однотипных элементов
- если матрица имеет имя `A`, то обращение `A[i, k]` обозначает элемент, расположенный на пересечении строки `i` и столбца `k`
- элементы, у которых номера строки и столбца совпадают, расположены на главной диагонали<sup>2</sup>

A[1,1]			
	A[2,2]		
		A[3,3]	
			A[4,4]

- выше главной диагонали расположены элементы, у которых номер строки **меньше** номера столбца:

	A[1,2]	A[1,3]	A[1,4]
		A[2,3]	A[2,4]
			A[3,4]

- ниже главной диагонали расположены элементы, у которых номер строки **больше** номера столбца:

A[2,1]			
A[3,1]	A[3,2]		
A[4,1]	A[4,2]	A[4,3]	

### Пример задания:

Дан фрагмент программы, обрабатывающей двухмерный массив `A` размера `n`×`n`.

```
k := 1;
for i:=1 to n do begin
  c := A[i,i];
  A[i,i] := A[k,i];
  A[k,i] := c;
end
```

Представим массив в виде квадратной таблицы, в которой для элемента массива `A[i,j]` величина `i` является номером строки, а величина `j` – номером столбца, в котором расположен элемент. Тогда данный алгоритм меняет местами

- 1) два столбца в таблице
- 2) две строки в таблице

<sup>1</sup> Здесь рассматривается только язык Паскаль, который является наиболее распространенным в школах России.

<sup>2</sup> В этом примере используется стандартная нумерация для Паскаля: индексы начинаются с единицы.

- 3) элементы диагонали и k-ой строки таблицы
- 4) элементы диагонали и k-го столбца таблицы

**Решение:**

- 1) сначала разберемся, что происходит внутри цикла; легко проверить (хотя бы ручной прокруткой, если вы сразу не узнали стандартный алгоритм), что операторы

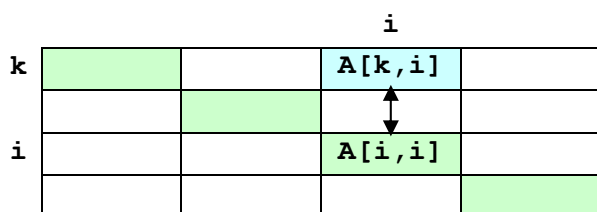
```

с := A[i, i];
A[i, i] := A[k, i];
A[k, i] := с;

```

меняют местами значения  $A[i, i]$  и  $A[k, i]$ , используя переменную  $c$  в качестве вспомогательной ячейки;

- 2) элемент матрицы  $A[i, i]$ , у которого номера строки и столбца одинаковые, стоит на главной диагонали; элемент  $A[k, i]$  стоит в том же столбце  $i$ , но в строке с номером  $k$ ; это значит, что в столбце  $i$  меняются местами элемент на главной диагонали и элемент в строке  $k$



- 3) так как эти операторы находятся в цикле, где переменная  $i$  принимает последовательно все значения от 1 до  $n$ , обмен выполняется для всех столбцов матрицы; то есть, все элементы главной диагонали меняются с соответствующими элементами строки  $k$
- 4) перед циклом стоит оператор присваивания  $k := 1$ ; а после него переменная  $k$  не меняется; поэтому в программе элементы главной диагонали обмениваются с первой строкой
- 5) таким образом, правильный ответ – 3.

**Возможные ловушки и проблемы:**

- сложность этой задачи в том, что все действия нужно «прокручивать в уме» (или на бумаге), не используя компьютер для отладки
- главная проблема – не перепутать столбцы и строки; номер строки – это (по соглашению) первый индекс элемента матрицы, а номер столбца – второй

**Совет:**

- чтобы понять, что делает программа, часто бывает полезно сделать ручную прокрутку на матрице небольшого размера, например, 3 на 3 или 4 на 4.
- если матрица небольшая (скажем, 5 на 5) можно (а иногда и нужно) вообще сделать все вычисления вручную и посмотреть, что получится

**Еще пример задания:**

Значения двух массивов  $A[1..100]$  и  $B[1..100]$  задаются с помощью следующего фрагмента программы:

```

for n:=1 to 100 do
  A[n] := (n-80) * (n-80);
for n:=1 to 100 do
  B[101-n] := A[n];

```

Какой элемент массива  $B$  будет наибольшим?

- 1)  $B[1]$
- 2)  $B[21]$
- 3)  $B[80]$
- 4)  $B[100]$

**Решение:**

- 1) здесь два цикла, в первом из них заполняется массив **A**, во втором – массив **B**
- 2) в элемент массива **A[n]** записывается квадрат числа **n-80**; все элементы массива **A** неотрицательны (как квадраты чисел)
- 3) посмотрим чему равны некоторые элементы массива **A**:  
 $A[1] = (1-80)^2 = (-79)^2 = 79^2$        $A[2] = (2-80)^2 = (-78)^2 = 78^2$   
...  
 $A[80] = (80-80)^2 = (0)^2 = 0$        $A[81] = (81-80)^2 = (1)^2 = 1$   
...  
 $A[99] = (99-80)^2 = 19^2$        $A[100] = (100-80)^2 = 20^2$
- 4) таким образом, при увеличении **n** от 1 до 80 значение **A[n]** уменьшается от  $79^2$  до нуля, а потом (для  $n > 80$ ) возрастает до  $20^2$
- 5) отсюда следует, что максимальное значение в массиве **A** – это  $A[1] = 79^2$
- 6) во втором цикле для всех номеров **n** от 1 до 100 выполняется оператор

```
B[101-n] := A[n];
```

который просто переписывает элементы массива **A** в массив **B**, «развертывая» массив в обратном порядке (элемент **A[1]** будет записан в **B[100]**, а **A[100]** – в **B[1]**)

- 7) **A[1]**, наибольший элемент массива **A**, будет записан в **B[100]**, поэтому **B[100]** – наибольший элемент в массиве **B**
- 8) таким образом, правильный ответ – **4**.

**Еще пример задания:**

*Значения элементов двумерного массива  $A[1..10,1..10]$  задаются с помощью следующего фрагмента программы:*

```
for i:=1 to 10 do  
  for k:=1 to 10 do  
    if i > k then  
      A[i,k] := 1  
    else A[i,k] := 0;
```

*Чему равна сумма элементов массива после выполнения этого фрагмента программы?*

- 1) 45                      2) 50                      3) 90                      4) 100

**Решение:**

- 1) в программе есть вложенный цикл, в котором переменная **i** обозначает строку, а **k** – столбец матрицы
- 2) элементы, для которых **i=k** – это главная диагональ матрицы, поэтому элементы, для которых **i > k** (только они будут равны 1), находятся под главной диагональю
- 3) в первой строке единичных элементов нет, во второй есть один такой элемент, в третьей – 2, в последней (10-ой) их 9, поэтому сумма элементов массива равна  
 $1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 = 45$
- 4) таким образом, правильный ответ – **1**.
- 5) при большом размере массива (например, 100 на 100) суммирование может оказаться трудоемким, поэтому лучше вспомнить формулу для вычисления суммы элементов арифметической прогрессии (именно такая прогрессия у нас, с шагом 1):

$$S = N \cdot \frac{a_1 + a_N}{2},$$

где  $N$  - количество элементов, а  $a_1$  и  $a_N$  – соответственно первый и последний элементы последовательности; в данном случае имеем

$$S = 9 \cdot \frac{1+9}{2} = 45.$$

- б) если приведенная выше формула прочно забыта, можно попытаться сгруппировать слагаемые в пары с равной суммой (как сделал, будучи школьником, великий математик К.Ф. Гаусс), например:

$$1 + 2 + \dots + 9 = (1+9) + (2+8) + \dots + (4+6) + 5 = 4 \cdot 10 + 5$$

### Еще пример задания:

Значения элементов двумерного массива  $A[1..10, 1..10]$  сначала равны 5. Затем выполняется следующий фрагмент программы:

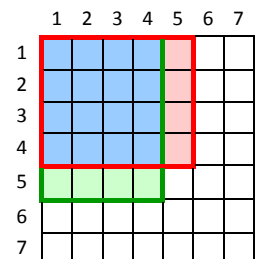
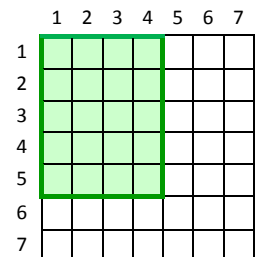
```
for i:=1 to 5 do
  for j:=1 to 4 do begin
    A[i,j]:=A[i,j]+5; { 1 }
    A[j,i]:=A[j,i]+5; { 2 }
  end;
```

Сколько элементов массива будут равны 10?

- 1) 8                      2) 16                      3) 24                      4) 0

#### Решение (вариант 1, анализ алгоритма):

- обратим внимание, что в двойном цикле переменная  $i$  изменяется от 1 до 5, а  $j$  – от 1 до 4 (на 1 шаг меньше)
- внутри цикла в операторе, отмеченном цифрой 1 в комментарии, в записи  $A[i, j]$  переменная  $i$  – это строка, а  $j$  – столбец, поэтому по одному разу обрабатываются все элементы массива, выделенные зеленым цветом:
- это значит, что если оставить только один первый оператор внутри цикла, все выделенные элементы увеличиваются на 5 и станут равны 10
- теперь рассмотрим второй оператор внутри цикла: в записи  $A[j, i]$  переменная  $i$  – это столбец, а  $j$  – строка, поэтому по одному разу обрабатываются (увеличиваются на 5) все элементы массива, выделенные рамкой красного цвета на рисунке справа
- теперь хорошо видно, что левый верхний угол массива (квадрат 4 на 4, синяя область) попадает в обе области, то есть, эти 16 элементов будут дважды увеличены на 5: они станут равны 15 после выполнения программы
- элементы, попавшие в зеленый и красный «хвостики» обрабатываются (увеличиваются на 5) по одному разу, поэтому они-то и будут равны 10
- всего таких элементов – 8 штук
- таким образом, правильный ответ – 1.



#### Решение (вариант 2, прокрутка небольшого массива):

- понятно, что в программе захватывается только левый верхний угол массива, остальные элементы не меняются
- сократим размер циклов так, чтобы можно было легко выполнить программу вручную; при этом нужно сохранить важное свойство: внутренний цикл должен содержать на 1 шаг меньше, чем внешний:

```
for i:=1 to 3 do
  for j:=1 to 2 do begin
```

```

A[i,j]:=A[i,j]+5; { 1 }
A[j,i]:=A[j,i]+5; { 2 }
end;

```

3) выполняя вручную этот вложенный цикл, получаем

	1	2	3	4	5
1	15	15	10	5	5
2	15	15	10	5	5
3	10	10	5	5	5
4	5	5	5	5	5
5	5	5	5	5	5

- 4) видим, что в самом углу получился квадрат 2 на 2 (по количеству шагов внутреннего цикла), в котором все элементы равны 15; по сторонам этого квадрата стоят 4 элемента, равные 10, их количество равно удвоенной стороне квадрата
- 5) в исходном варианте внутренний цикл выполняется 4 раза, поэтому угловой квадрат будет иметь размер 4 на 4; тогда 8 элементов, граничащих с его сторонами, будут равны 10
- 6) таким образом, правильный ответ – 1.

**Возможные проблемы:**

- упрощая задачу, нельзя потерять ее существенные свойства: например, здесь было важно, что внутренний цикл содержит на 1 шаг меньше, чем внешний

### Задачи для тренировки<sup>3</sup>:

- 1) Значения двух массивов A[1..100] и B[1..100] задаются с помощью следующего фрагмента программы:

```
for n:=1 to 100 do
  A[n] := n - 10;
for n:=1 to 100 do
  B[n] := A[n]*n;
```

Сколько элементов массива B будут иметь положительные значения?

- 1) 10                      2) 50                      3) 90                      4) 100
- 2) Все элементы двумерного массива A размером 10x10 элементов первоначально были равны 0. Затем значения элементов меняются с помощью вложенного оператора цикла в представленном фрагменте программы:

```
for n:=1 to 4 do
  for k:=n to 4 do begin
    A[n,k] := A[n,k] + 1;
    A[k,n] := A[k,n] + 1;
  end;
```

Сколько элементов массива в результате будут равны 1?

- 1) 0                      2) 16                      3) 12                      4) 4
- 3) Значения двумерного массива задаются с помощью вложенного оператора цикла в представленном фрагменте программы:

```
for n:=1 to 5 do
  for k:=1 to 5 do
    B[n,k] := n + k;
```

Чему будет равно значение B(2,4)?

- 1) 9                      2) 8                      3) 7                      4) 6
- 4) Дан фрагмент:

```
for n:=1 to 6 do
  for m:=1 to 5 do begin
    C[n,m] :=C[n,m] + (2*n-m) ;
  end;
```

---

<sup>3</sup> Источники заданий:

1. Демонстрационные варианты ЕГЭ 2004-2009 гг.
2. Гусева И.Ю. ЕГЭ. Информатика: раздаточный материал тренировочных тестов. — СПб: Тригон, 2009.
3. Якушкин П.А., Крылов С.С. ЕГЭ-2010. Информатика: сборник экзаменационных заданий. — М.: Эксмо, 2009.
4. Якушкин П.А., Лещинер В.Р., Кириенко Д.П. ЕГЭ 2010. Информатика. Типовые тестовые задания. — М.: Экзамен, 2010.
5. 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.
6. Крылов С.С., Ушаков Д.М. ЕГЭ 2010. Информатика. Тематическая рабочая тетрадь. — М.: Экзамен, 2010.
7. Якушкин П.А., Ушаков Д.М. Самое полное издание типовых вариантов реальных заданий ЕГЭ 2010. Информатика. — М.: Астрель, 2009.
8. Абрамян М.Э., Михалкович С.С., Русанова Я.М., Чердынцева М.И. Информатика. ЕГЭ шаг за шагом. — М.: НИИ школьных технологий, 2010.
9. Чуркина Т.Е. ЕГЭ 2011. Информатика. Тематические тренировочные задания. — М.: Эксмо, 2010.

Чему будет равно значение  $C[4,3]$ , если перед этими командами значение  $C[4,3]=10$ ?

- 1) 5                      2) 10                      3) 15                      4) 25

- 5) Значения элементов двух массивов A и B размером 1 x 100 задаются с помощью следующего фрагмента программы:

```
for i:=1 to 100 do
  A[i] := 50 - i;
for i:=1 to 100 do
  B[i] := A[i] + 49;
```

Сколько элементов массива B будут иметь отрицательные значения?

- 1) 1                      2) 10                      3) 50                      4) 100

- 6) Значения элементов двумерного массива A были равны 0. Затем значения некоторых элементов были изменены (см. представленный фрагмент программы):

```
n := 0;
for i:=1 to 5 do
  for j:=1 to 6-i do begin
    n := n + 1;
    A[i,j] := n;
  end;
```

Какой элемент массива будет иметь в результате максимальное значение?

- 1) A[1,1]                      2) A[1,5]                      3) A[5,1]                      4) A[5,5]

- 7) Значения элементов двумерного массива A размером 5x5 задаются с помощью вложенного цикла в представленном фрагменте программы:

```
for i:=1 to 5 do
  for j:=1 to 5 do begin
    A[i,j] := i*j;
  end;
```

Сколько элементов массива будут иметь значения больше 10?

- 1) 12                      2) 8                      3) 10                      4) 4

- 8) Значения элементов двумерного массива A размером 5x5 задаются с помощью вложенного цикла в представленном фрагменте программы:

```
for i:=1 to 5 do
  for j:=1 to 5 do begin
    A[i,j] := i + j;
  end;
```

Сколько элементов массива будут иметь значения больше 5?

- 1) 5                      2) 20                      3) 10                      4) 15

- 9) Дан фрагмент программы:

```
for n:=1 to 5 do
  for m:=1 to 5 do
    C[n,m] := (m - n) * (m - n);
```

Сколько элементов массива С будут равны 1?

- 1) 5                      2) 2                      3) 8                      4) 14

10) В программе описан одномерный целочисленный массив с индексами от 0 до 10. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do
  A[i]:= i + 1;
for i:=1 to 10 do
  A[i]:= A[i-1];
```

Как изменяются элементы этого массива?

- 1) все элементы, кроме последнего, сдвигаются на 1 элемент вправо  
2) все элементы, кроме первого, сдвигаются на 1 элемент влево  
3) все элементы окажутся равны 1  
4) все элементы окажутся равны своим индексам

11) В программе описан одномерный целочисленный массив с индексами от 0 до 10. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do
  A[i]:= i + 1;
for i:=10 downto 0 do
  A[i]:= A[10-i];
```

Чему будут равны элементы этого массива?

- 1) 10 9 8 7 6 5 4 3 2 1 0  
2) 11 10 9 8 7 6 5 4 3 2 1  
3) 11 10 9 8 7 6 7 8 9 10 11  
4) 1 2 3 4 5 6 5 4 3 2 1

12) В программе описан одномерный целочисленный массив с индексами от 0 до 10. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do
  A[i]:= i + 1;
for i:=0 to 10 do
  A[i]:= A[10-i];
```

Чему будут равны элементы этого массива?

- 1) 10 9 8 7 6 5 4 3 2 1 0  
2) 11 10 9 8 7 6 5 4 3 2 1  
3) 11 10 9 8 7 6 7 8 9 10 11  
4) 10 9 8 7 6 5 6 7 8 9 10

13) В программе описан одномерный целочисленный массив с индексами от 0 до 10. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do
  A[i]:= i - 1;
for i:=1 to 10 do
  A[i-1]:= A[i];
A[10] := 10;
```

Как изменяются элементы этого массива?

- 1) все элементы, кроме последнего, окажутся равны между собой



- 2) все элементы окажутся равны своим индексам
- 3) все элементы, кроме последнего, сдвигаются на один элемент вправо
- 4) все элементы, кроме последнего, уменьшаются на единицу

14) В программе описан одномерный целочисленный массив с индексами от 0 до 10. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do
  A[i]:= i;
for i:=1 to 11 do
  A[i-1]:= A[11-i];
```

Чему будут равны элементы этого массива?

- 1) 10 9 8 7 6 5 4 3 2 1 0
- 2) 11 10 9 8 7 6 5 4 3 2 1
- 3) 10 9 8 7 6 5 6 7 8 9 10
- 4) 11 10 9 8 7 6 7 8 9 10 11

15) В программе описан одномерный целочисленный массив с индексами от 0 до 10. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do A[i]:=i;
for i:=0 to 10 do begin
  A[10-i]:=A[i];
  A[i]:=A[10-i];
end;
```

Чему будут равны элементы этого массива?

- 1) 10 9 8 7 6 5 4 3 2 1 0
- 2) 0 1 2 3 4 5 6 7 8 9 10
- 3) 10 9 8 7 6 5 6 7 8 9 10
- 4) 0 1 2 3 4 5 4 3 2 1 0

16) Элементы двумерного массива A размером N×N первоначально были равны 1000. Затем значения некоторых из них меняют с помощью следующего фрагмента программы:

```
k := 0;
for i:=1 to N do
  for j:=N-i+1 to N do begin
    k:= k + 1;
    A[i,j]:= k;
  end;
```

Какой элемент массива в результате будет иметь минимальное значение?

- 1) A[1,1]    2) A[1,N]    3) A[N,1]    4) A[N,N]

17) Элементы двумерного массива A размером 9×9 задаются с помощью следующего фрагмента программы:

```
for n:=1 to 9 do
  for k:=1 to 9 do
    A[n,k]:=n+k+1;
```

Сколько элементов массива A будут принимать четные значения?

- 1) 36            2) 40            3) 41            4) 45

18) Значения элементов двух массивов A[1..100] и B[1..100] задаются с помощью следующего фрагмента программы:

```
for n:=1 to 100 do
  A[n] := n - 50;
for n:=1 to 100 do
  B[101-n] := A[n]*A[n];
```

Какой элемент массива B будет наименьшим?

- 1) B[1]      2) B[50]      3) B[51]      4) B[100]

19) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленные переменные k, i. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do A[i]:=i;
for i:=10 downto 0 do begin
  k:=A[10-i];
  A[10-i]:=A[i];
  A[i]:=k;
end;
```

Чему будут равны элементы этого массива?

- 1) 10 9 8 7 6 5 4 3 2 1 0  
2) 0 1 2 3 4 5 6 7 8 9 10  
3) 10 9 8 7 6 5 6 7 8 9 10  
4) 0 1 2 3 4 5 4 3 2 1 0

20) Элементы двумерного массива A размером 4×4 первоначально были равны 0. Затем они изменяются с помощью следующего фрагмента программы:

```
for n:=1 to 4 do
  for k:=n to 4 do
    A[n,k] := 1;
```

Сколько элементов массива A будут равны 1?

- 1) 4      2) 8      3) 10      4) 16

21) Элементы двумерного массива A размером 10×10 первоначально были равны 1. Затем значения некоторых из них меняют с помощью следующего фрагмента программы:

```
for n:=1 to 4 do
  for k:=1 to n+1 do begin
    A[n,k] := A[n,k] - 1;
    A[n,k+1] := A[n,k] - 1;
  end;
```

Сколько элементов массива в результате будут равны 0?

- 1) 0      2) 4      3) 8      4) 16

22) Дан фрагмент программы, обрабатывающий массив A из 10 элементов:

```
n := 10;
for i:=1 to n do A[i] := i;
j := 1;
for i:=1 to n-1 do
  if A[i] < A[i+1] then j := j + 1;
```

Чему будет равно значение переменной  $j$  после выполнения этого алгоритма?

- 1) 1                      2) 2                      3) 10                      4) 11

23) Значения элементов двумерного массива  $A[1..100,1..100]$  задаются с помощью следующего фрагмента программы:

```
for i:=1 to 100 do
for k:=1 to 100 do
if i = k then
    A[i,k] := 1
else A[i,k] := -1;
```

Чему равна сумма элементов массива после выполнения этого фрагмента программы?

- 1) 0                      2) -9800                      3) -9900                      4) -10000

24) Значения элементов двумерного массива  $A[1..100,1..100]$  задаются с помощью следующего фрагмента программы:

```
for i:=1 to 100 do
for k:=1 to 100 do
if i > k then
    A[i,k] := 1
else A[i,k] := -1;
```

Чему равна сумма элементов массива после выполнения этого фрагмента программы?

- 1) 0                      2) 100                      3) -100                      4) -200

25) Значения элементов двумерного массива  $A[1..100,1..100]$  задаются с помощью следующего фрагмента программы:

```
for i:=1 to 100 do
for k:=1 to 100 do
if i > k then
    A[i,k] := i
else A[i,k] := -k;
```

Чему равна сумма элементов массива после выполнения этого фрагмента программы?

- 1) 5000                      2) 0                      3) -5000                      4) -5050

26) Дан фрагмент программы, обрабатывающий массив  $A$  из 10 элементов:

```
j := 1;
for i:=1 to 10 do
    if A[i] = A[j] then j := i;
s := j;
```

Чему будет равно значение переменной  $s$  после выполнения этого алгоритма?

- 1) 1  
2) 10  
3) индексу элемента, равного первому, и имеющему наибольший индекс  
4) индексу элемента, равного последнему, и имеющему наименьший индекс

27) Значения элементов двумерного массива  $A[1..10,1..10]$  сначала равны 4. Затем выполняется следующий фрагмент программы:

```
for i:=1 to 6 do
```

```

for j:=1 to 5 do begin
  A[i,j]:=A[i,j]+6;
  A[j,i]:=A[j,i]+6;
end;

```

Сколько элементов массива будут равны 10?

- 1) 30            2) 25            3) 10            4) 0

- 28) Значения элементов двумерного массива A[1..10,1..10] сначала равны 4. Затем выполняется следующий фрагмент программы:

```

for i:=1 to 4 do
  for j:=1 to 5 do begin
    A[i,j]:=A[i,j]+4;
    A[j,i]:=A[j,i]+5;
  end;

```

Сколько элементов массива будут равны 9?

- 1) 20            2) 16            3) 5            4) 4

- 29) Значения элементов двумерного массива A[1..10,1..10] сначала равны 0. Затем выполняется следующий фрагмент программы:

```

for i:=1 to 4 do
  for j:=2 to 5 do begin
    A[i,j]:=A[i,j]+4;
    A[j,i]:=A[j,i]+5;
  end;

```

Сколько элементов массива будут равны 9?

- 1) 20            2) 16            3) 9            4) 4

- 30) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленные переменные **k**, **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```

for i:=0 to 10 do A[i]:=i;
for i:=0 to 4 do begin
  k:=A[i];
  A[i]:=A[10-i];
  A[10-i]:=k;
end;

```

Чему будут равны элементы этого массива?

- 1) 10 9 8 7 6 5 4 3 2 1 0  
 2) 0 1 2 3 4 5 6 7 8 9 10  
 3) 0 1 2 3 4 5 4 3 2 1 0  
 4) 10 9 8 7 6 5 6 7 8 9 10

- 31) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленные переменные **k**, **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```

for i:=0 to 10 do A[i]:=i;
for i:=0 to 10 do begin
  k:=A[i];
  A[i]:=A[10-i];
  k:=A[10-i];

```

**end;**

Чему будут равны элементы этого массива?

- 1) 10 9 8 7 6 5 4 3 2 1 0
- 2) 0 1 2 3 4 5 6 7 8 9 10
- 3) 0 1 2 3 4 5 4 3 2 1 0
- 4) 10 9 8 7 6 5 6 7 8 9 10

32) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленные переменные **k**, **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do A[i]:=i;  
for i:=0 to 4 do begin  
  k:=A[10-i];  
  A[10-i]:=A[i];  
  k:=A[i];  
end;
```

Чему будут равны элементы этого массива?

- 1) 10 9 8 7 6 5 4 3 2 1 0
- 2) 0 1 2 3 4 5 6 7 8 9 10
- 3) 0 1 2 3 4 5 4 3 2 1 0
- 4) 10 9 8 7 6 5 6 7 8 9 10

33) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленная переменная **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do A[i]:=i;  
for i:=0 to 9 do begin  
  A[i]:=A[i+1];  
end;
```

Чему будут равны элементы этого массива?

- 1) 10 0 1 2 3 4 5 6 7 8 9
- 2) 1 2 3 4 5 6 7 8 9 10 10
- 3) 0 0 1 2 3 4 5 6 7 8 9
- 4) 1 2 3 4 5 6 7 8 9 10 0

34) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленные переменные **k**, **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do A[i]:=i;  
k:=A[10];  
for i:=0 to 9 do  
  A[i]:=A[i+1];  
A[0]:=k;
```

Чему будут равны элементы этого массива?

- 1) 10 0 1 2 3 4 5 6 7 8 9
- 2) 10 2 3 4 5 6 7 8 9 10 10
- 3) 1 2 3 4 5 6 7 8 9 10 10
- 4) 1 2 3 4 5 6 7 8 9 10 0

- 35) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленные переменные **k**, **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do A[i]:=i;
for i:=0 to 4 do begin
  k:=A[2*i];
  A[2*i]:=A[2*i+1];
  A[2*i+1]:=k;
end;
```

Чему будут равны элементы этого массива?

- 1) 5 6 7 8 9 0 1 2 3 4 10
- 2) 10 9 8 7 6 5 4 3 2 1 0
- 3) 0 1 2 3 4 5 4 3 2 1 0
- 4) 1 0 3 2 5 4 7 6 9 8 10

- 36) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленные переменные **k**, **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do A[i]:=i;
for i:=0 to 4 do begin
  k:=A[i];
  A[i]:=A[i+5];
  A[i+5]:=k;
end;
```

Чему будут равны элементы этого массива?

- 1) 5 6 7 8 9 0 1 2 3 4 10
- 2) 10 9 8 7 6 5 4 3 2 1 0
- 3) 0 1 2 3 4 5 4 3 2 1 0
- 4) 1 0 3 2 5 4 7 6 9 8 10

- 37) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленные переменные **k**, **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do A[10-i]:=i;
k:=A[9];
for i:=0 to 9 do
  A[10-i]:=A[9-i];
A[1]:=k;
```

Чему будут равны элементы этого массива?

- 1) 9 8 7 6 5 4 3 2 1 0 10
- 2) 10 1 9 8 7 6 5 4 3 2 1
- 3) 0 10 9 8 7 6 5 4 3 2 1
- 4) 10 0 9 8 7 6 5 4 3 2 1

- 38) В программе описан одномерный целочисленный массив с индексами от 0 до 10 и целочисленная переменная **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 10 do A[i]:=i+1;
```

```
for i:=1 to 10 do
  A[i]:=A[i-1];
```

Чему будут равны элементы этого массива?

- 1) 1 1 2 3 4 5 6 7 8 9 10
- 2) 1 2 3 4 5 6 7 8 9 10 10
- 3) 1 1 1 1 1 1 1 1 1 1 1
- 4) 0 1 2 3 4 5 6 7 8 9 10

39) В программе описан одномерный целочисленный массив с индексами от 0 до 9 и целочисленные переменные **k** и **i**. В приведенном ниже фрагменте программы массив сначала заполняется, а потом изменяется:

```
for i:=0 to 9 do A[i]:=i+1;
k:=A[9];
for i:=9 downto 1 do
  A[i]:=A[i-1];
A[0]:=k;
```

Чему будут равны элементы этого массива?

- 1) 10 3 4 5 6 7 8 9 10 10
- 2) 10 1 2 3 4 5 6 7 8 9
- 3) 10 1 1 1 1 1 1 1 1 1
- 4) 10 10 10 10 10 10 10 10 10 10

40) В программе описан одномерный целочисленный массив **A** с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=0 to 10 do
  A[i]:=i-1;
for i:=10 downto 1 do
  A[i-1]:=A[i];
```

Чему будут равны элементы этого массива?

- 1) 9 9 9 9 9 9 9 9 9 9 9
- 2) 0 1 2 3 4 5 6 7 8 9 9
- 3) 0 1 2 3 4 5 6 7 8 9 10
- 4) -1 -1 0 1 2 3 4 5 6 7 8

41) В программе описан одномерный целочисленный массив **A** с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=0 to 10 do
  A[i]:=i;
for i:=0 to 5 do begin
  A[10-i]:=A[9-i];
  A[i]:=A[i+1];
end;
```

Чему будут равны элементы этого массива?

- 1) 0 1 2 3 4 5 6 7 8 9 10
- 2) 0 1 2 3 4 5 6 7 8 9 9
- 3) 1 2 3 4 5 5 6 7 8 9

4) 1 2 3 4 5 6 5 4 3 2 1

42) В программе обрабатывается двумерный целочисленный массив  $A [0..n,0..n]$ . Первый индекс элемента обозначает номер строки, а второй – номер столбца. Дан фрагмент программы:

```
for i:=0 to n do begin
  c:=A[i,n-i];
  A[i,n-i]:=A[1,i];
  A[1,i]:=c;
end;
```

Что меняет этот фрагмент программы?

- 1) два столбца в таблице
- 2) строку и столбец в таблице
- 3) элементы диагонали и строки в таблице
- 4) элементы диагонали и столбца в таблице

43) В программе описан одномерный целочисленный массив  $A$  с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=0 to 10 do
  A[i]:=i;
for i:=0 to 5 do begin
  A[5-i]:=A[5+i];
  A[2+i]:=A[10-i];
end;
```

Чему будут равны элементы этого массива?

- 1) 10 9 8 7 8 7 6 7 8 9 10
- 2) 9 8 7 6 5 9 8 7 6 5 10
- 3) 10 9 8 7 6 10 9 8 7 6 10
- 4) 10 9 8 7 6 5 6 7 8 9 10

44) В программе описан двухмерный целочисленный массив  $A [1..6,1..6]$ . Ниже представлен фрагмент этой программы, в котором изменяются значения элементов массива.

```
for n:=1 to 6 do
  for m:=1 to 6 do
    A[n,m]:=A[m,n]+2*n-m;
```

До выполнения данного фрагмента программы значение  $A[4,3]$  было равно 10, а значение  $A[3,4]$  было равно 15. Чему будет равно значение  $A[4,3]$  после выполнения этого фрагмента программы?

- 1) 10
- 2) 15
- 3) 17
- 4) 20

45) В программе описан одномерный целочисленный массив  $A$  с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=0 to 10 do
  A[i]:=10-i;
for i:=0 to 5 do begin
  A[10-i]:=A[5-i];
  A[5+i]:=A[i];
```



**end;**

Чему будут равны элементы этого массива?

- 1) 9 8 7 6 5 10 9 8 7 6 10
- 2) 10 9 8 7 6 5 6 7 8 9 10
- 3) 10 9 8 7 6 10 9 8 7 6 10
- 4) 5 6 7 8 9 10 9 8 7 6 5

46) В программе описан одномерный целочисленный массив с индексами от 0 до n. Известно, что в массиве есть несколько элементов с максимальным значением. Дан фрагмент программы:

```
j:=1;  
for i:=1 to n do  
  if A[i] > A[j] then j:= i;  
s:=j;
```

Чему будет равно значение переменной **s** после выполнения этого фрагмента программы:

- 1) значению максимального элемента
- 2) количеству элементов в массиве A, имеющих максимальное значение
- 3) индексу первого элемента в массиве A, имеющего максимальное значение
- 4) индексу последнего элемента в массиве A, имеющего максимальное значение

47) Дан фрагмент программы, обрабатывающий двухмерный массив A размером n×n.

```
for i:=1 to n-1 do  
  for j:=1 to n do  
    if A[i,1] < A[j,1] then begin  
      k:=A[i,1];  
      A[i,1]:=A[j,1];  
      A[j,1]:=k;  
    end;
```

В этом фрагменте:

- 1) упорядочивается первая строка массива по убыванию
- 2) упорядочивается первый столбец массива по убыванию
- 3) заменяются элементы k-ого столбца таблицы
- 4) заменяются элементы k-ой строки таблицы

48) Значения двух массивов A и B с индексами от 1 до 100 задаются при помощи следующего фрагмента программы:

```
for i:=1 to n do  
  A[i]:=i*i;  
for i:=1 to n do  
  B[i]:=A[i]-100;
```

Сколько положительных значений будет в массиве B?

- 1) 0
- 2) 10
- 3) 90
- 4) 91

49) Значения двух массивов A и B с индексами от 1 до 100 задаются при помощи следующего фрагмента программы:

```
for i:=1 to n do  
  A[i]:=(i-75)*(i-75);  
for i:=1 to n do
```

```
B[101-i]:=A[i];
```

Какой элемент массива В будет наибольшим?

- 1) В[1]          2) В[26]          3) В[75]          4) В[100]

50) Значения двумерного массив А размером 9×9 задаются при помощи следующего фрагмента программы:

```
for n:=1 to 9 do  
  for k:=1 to 9 do  
    A[n,k]:=n+k+1;
```

Сколько четных значений будет в массиве А?

- 1) 36          2) 40          3) 41          4) 45

51) В программе описан одномерный целочисленный массив А с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=0 to 10 do  
  A[i]:=2+i;  
for i:=0 to 4 do begin  
  A[i]:=A[10-i]-1;  
  A[10-i]:=A[i]+3;  
end;
```

Чему будут равны элементы этого массива?

- 1) 11 10 9 8 7 5 6 7 8 9  
2) 11 10 9 8 7 7 5 6 7 8 9  
3) 11 10 9 8 7 7 10 11 12 13 14  
4) 11 10 9 8 7 10 11 12 13 14

52) В программе описан одномерный целочисленный массив А с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=0 to 10 do  
  A[i]:=2+i;  
for i:=0 to 4 do begin  
  A[10-i]:=2*A[10-i];  
  A[i]:=A[i+1]+4;  
end;
```

Чему будут равны элементы этого массива?

- 1) 7 8 9 10 11 7 16 18 20 22 24  
2) 7 8 9 10 11 16 18 20 22 24  
3) 2 3 4 5 6 7 16 18 20 22 24  
4) 3 4 5 6 7 7 16 18 20 22 24

53) В программе описан одномерный целочисленный массив А с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=0 to 10 do  
  A[i]:=2*i;  
for i:=0 to 4 do begin
```

```

A[10-i]:=A[i]-1;
A[i]:=A[10-i]-1;
end;

```

Чему будут равны элементы этого массива?

- 1) 0 2 4 6 8 10 12 14 16 18 20
- 2) 19 17 15 13 11 10 -1 1 3 5 7
- 3) -2 0 2 4 6 10 7 5 3 1 -1
- 4) -1 1 3 5 7 9 11 13 15 17 19

54) В программе описан одномерный целочисленный массив A с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```

A[0]:=1;
for i:=1 to 10 do
  A[i]:=2+A[i-1];
for i:=0 to 5 do
  A[10-i]:=A[i]-1;

```

Чему будут равны элементы этого массива?

- 1) 1 3 5 7 9 10 8 6 4 2 0
- 2) 1 3 5 7 9 11 13 15 17 19 21
- 3) 1 3 5 7 9 11 9 7 5 3 1
- 4) 1 3 5 7 9 11 8 6 4 2 0

55) В программе описан одномерный целочисленный массив A с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```

for i:=0 to 10 do
  A[i]:=i;
t:=A[0];
for i:=1 to 10 do
  A[i-1]:=A[i];
A[10]:=t;

```

Чему будут равны элементы этого массива?

- 1) 10 10 10 10 10 10 10 10 10 10
- 2) 1 2 3 4 5 6 7 8 9 10 0
- 3) 0 0 0 0 0 0 0 0 0 0
- 4) 1 2 3 4 5 6 7 8 9 10 1

56) В программе описан одномерный целочисленный массив A с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```

for i:=0 to 10 do
  A[i]:=i+3;
for i:=10 downto 0 do begin
  k:=A[i];
  A[i]:=A[10-i];
  A[10-i]:=k;
end;

```

Чему будут равны элементы этого массива?

- 1) 13 12 11 10 9 8 7 6 5 4 3
- 2) 3 4 5 6 7 8 9 10 11 12 13
- 3) 13 12 11 10 9 8 9 10 11 12 13
- 4) 3 4 5 6 7 8 7 6 5 4 3

57) Дан фрагмент программы, обрабатывающей двумерный массив A[1..5,1..4]:

```
k:=4;  
for m:=1 to 4 do begin  
  k:=k+1;  
  for n:=1 to 5 do begin  
    k:=m-k;  
    A[n,m] :=n*n+m*m-2*k;  
  end;  
end;
```

Чему будет равно значение A[3,1]?

- 1) 18
- 2) 19
- 3) 20
- 4) 21

58) В программе описан одномерный целочисленный массив A с индексами от 1 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=1 to 10 do  
  A[i]:=2*i;  
for i:=1 to 10 do begin  
  k:=2*A[i]+3;  
  A[10-i+1]:=k;  
end;
```

Чему будут равны элементы этого массива?

- 1) 7 11 15 19 23 27 31 35 39 43
- 2) 17 25 33 41 49 23 19 15 11 7
- 3) 5 9 13 17 21 25 29 33 37 41
- 4) 43 39 35 31 27 23 19 15 11 7

59) В программе описан одномерный целочисленный массив A с индексами от 1 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=1 to 10 do  
  A[i]:=5*i;  
for i:=1 to 10 do begin  
  k:=A[i]-2;  
  A[10-i+1]:=k;  
end;
```

Чему будут равны элементы этого массива?

- 1) 1 6 11 16 21 23 18 13 8 3
- 2) 3 8 13 18 23 28 33 38 43 48
- 3) 48 43 38 33 28 23 18 13 8 3
- 4) 1 6 11 16 21 26 31 36 41 46

60) В программе описан одномерный целочисленный массив A с индексами от 0 до 10. Ниже представлен фрагмент этой программы, в котором значения элементов массива сначала задаются, а затем меняются.

```
for i:=0 to 10 do
  A[i]:=3*i;
for i:=1 to 10 do
  A[i]:=A[i] mod 3;
```

Чему будут равны элементы этого массива?

- 1) Все элементы будут равны 3.
- 2) Все элементы будут равны 1.
- 3) Все элементы будут равны 0.
- 4) Все элементы будут равны своим индексам.