Регулярные типы данных. Одномерные массивы. Работа с элементами

Урок 8

Тема урока: Одномерные массивы. Работа с элементами

- План урока:

1. Регулярные типы данных. Одномерные массивы. Работа с элементами.
2. Решение задач.
3. Домашнее задание.

- В математике, экономике, информатике часто используются упорядоченные наборы данных, например последовательности чисел, таблицы, списки фамилий. Для обработки наборов данных одного типа вводится понятие массива.

Массив - совокупность конечного числа данных одного типа.

- Массив обозначается одним именем.
- Всю совокупность действительных чисел 1.6, 14.9, -5.0, 8.5, 0.46 можно считать массивом и обозначить одним именем, например А.
- Каждый элемент массива обозначается именем массива с индексом. Элементы массива упорядочены по значениям индекса.

В математике, как правило, индекс либо заключается в круглые скобки, либо указывается несколько ниже имени массива, например: А(1), А(2), А(3), А(4), А(5) или А1, А2, А3, А4, А5, или в общем виде **{Ai}** , где i =1, 2, 3, ..., n

В языке Qbasic индекс заключается в круглые скобки.

В Turbo Pascal 7.0 в квадратные.

Для рассмотренного примера элементами массива А являются:

На QBasic A(1)=1.6, A(2)=14.9, A(3)= -5.0, A(4)=8.5, A(5)=0.46.

На Turbo Pascal 7.0 A[1]:=1.6, A[2]:=14.9, A[3]:= -5.0, A[4]:=8.5, A[5]:=0.46.

Таким образом - индекс определяет положение элемента массива данных относительно его начала.

Если в программе используется массив, то он должен быть описан.

Форма описания имеет вид

DIM имя массива (n1 TO n2) AS тип элементов

Var имя массива: ARRAY [n1 .. n2] OF <тип элементов>;

DIM A(1 TO 1000) AS INTEGER

Var a:array[1..1000] of integer;

Здесь n1, n2 — номер первого и последнего элементов массива соответственно.

В качестве типа элементов может использоваться любой тип данных, кроме файлового.

Если несколько массивов имеют одинаковый тип индексов и одинаковый базовый тип, то допускается в описании объединять массивы в список, например

DIM A(50, B(50), C(50) AS SINGLE

Var A, B, C: ARRAY [1..50] OF REAL;

Здесь объявлено списком три массива A, B, C действительных чисел, каждый из которых содержит по 50 элементов (от 1 до 50):

Нельзя путать понятия «индекс» и «тип индекса». Тип индекса - только в разделе описания массива, а индекс - в разделе операторов для обозначения конкретных элементов массива. При этом индекс должен быть того же типа, что и описание типа индекса.

В качестве индекса может быть выражение, частным случаем которого является константа или переменная. Элемент массива иначе называется переменной с индексом. В отличие от нее, переменная без индекса называется простой переменной.

Элементы массива могут стоять как в левой части оператора присваивания, так и в выражениях. Над элементами массива можно производить те же операции, которые допустимы для данных его базового типа. Если базовый тип есть INTEGER, то допустимы все операции над данными целого типа, включая и стандартные функции.

Присвоение значений элементам массива может происходить с помощью:

1. оператора INPUT: оператора Read (Readln);

uses crt;

DEFINT I,N

Var i,n:integer;

DIM A(10000) AS INTEGER

a:array[1..10000] of integer;

Begin

CLS

clrscr;

Определяем число элементов в массиве

INPUT “N=”;N

write('Введ. Размер. массива ');readln(n);

Ввод элементов массива

```
FOR I=1 TO N

INPUT A(I)

END
```

```
for i:=1 to n do
begin
write('a[',i,'] ');
readln(a[i]);
end;
```

Выводим на экран полученный массив

```
FOR I=1 TO N

PRINT A(I);”  “;
NEXT

END
```

```
for i:=1 to n do
begin
write(a[i],' ');
end;
readkey;
 end.
```

Присвоение значений элементам массива может происходить с помощью:

2. **Оператора DATA:**

описания элементов массива описания констант.

	uses crt;
CONST N=5	Const n=5;
DATA 4,5,7,7,0	a:array[1..5] of integer= (4,5,7,7,0);
DEFINT I	Var i:integer;
	Begin
CLS	clrscr;
Считываем и печатаем полученный массив	
FOR I=1 TO N	for i:=1 to n do begin
READ A(I):PRINT A(I);” “;	write(a[i],' ');
NEXT	end;
	readkey;
END	end.

3. оператора присвоения

(для небольших массивов):

A(1)=10: A(2)=0:A(3)=2 a[1]:=10; a[2]:=0, … a[N]:=2;

После того, как данные введены в массив, ими можно оперативно распоряжаться. Доступен любой элемент массива. Достаточно только указать его индекс.

Для получения случайных чисел воспользуемся следующей функцией

Функция RND [(x)] в QBasic

Функция RND [(x)] возвращает случайное число обычной точности в интервале от 0 до 1. Если аргумент опущен, возвращается следующее число из последовательности случайных чисел.

Для запуска генератора используйте **RANDOMIZE TIMER**.

Функция Random [(x)] в TPascal 7.0

Формирует случайное число от 0 до X целого или вещественного типа (перед обращением к функции ее целесообразно инициализировать, использовав процедуру Randomize).

X – параметр, указывающий диапазон значений случайного числа. Оно изменяется в пределах 0 до X. Результат в этом случае имеет тип Word (диапазон значений - 0…65535).

Если параметр X не задан, результат будет типа Real в пределах 0.0 <=X <1.0.

Для получения массива целых случайных чисел из диапазона от [A,B]

RND(B-A)+A random(B-A)+A

Сформировать одномерный массив из N элементов, где элементы массива – целые случайные числа в пределах от 1 до 45. Напечатать элементы массива в прямом и обратном порядке

```
‘Prim1

DEFINT I,N
DIM A(1 TO 10000) AS INTEGER

CLS
RANDOMIZE TIMER
INPUT” размер N=”;N

FOR I=1 TO N
```

```
Program Prim1;
uses crt;
Var i,n:integer;
a:array[1..10000] of integer;
Begin
clrscr;
 randomize;
write(' размер = ');readln(n);
 for i:=1 to n do
begin
```

Получаем случайные числа в пределах от 1 до 45.	
A(I)=RND*45+1	*a[i]:=random(46))+1;*
Выводим элементы массива на экран.	
PRINT A(I);” “;	*write(a[i],' ');*
NEXT	*end;*
Полученный массив печатаем в обратном порядке	
FOR I=N TO 1 STEP-1	*for i:=n down to 1 do begin*
PRINT A(I);” “;	*write(a[i],' ');*
NEXT	*end;*
	readkey;
END	*end.*

QB TP

В предыдущей задаче переставить элементы, стоящие на нечетных местах, с соответствующими элементами на четных местах.

После создания массива вставим строки:

В полученном массиве меняем соседние элементы.

I=1	**i:=1;**
	Пока I<=N-1
WHILE I<=N-1	**while i<=n-1 do begin**

Меняем значения соседних элементов.

R=A(I):A(I)=A(I+1):A(I+1)=R	**r:=a[i];a[i]:=a[i+1];a[i+1]:=r;**

Увеличиваем индекс на два.

I=I+2	**inc(i,2);**
WEND	**end;**

Распечатываем измененный массив.

FOR I=1 TO N	**for i:=1 to n do begin**
PRINT A(I);” “;	**write(a[i],' ');**
NEXT	**end;**
	readkey;
END	**end.**

QB TP

Найти максимальный (минимальный) элемента массива, а также его порядковый номер.

После создания массива вставим строки:

За начальный максимум (минимум) берем первый элемент массива.

MIN=A(1):MAX=A(1) **min:=a[1]; max:=a[1];**

FOR I=1 TO N **for i:=1 to n do begin**

Если найдется элемент меньший **MIN**, то **MIN** будет равен этому элементу.

Одновременно запоминаем индекс промежуточного минимума.

IF A(I)<=MIN THEN **If a[i]<=min then begin**

MIN=A(I):IMIN=I **Min:=a[i]; imin:=i; end;**

Если найдется элемент больший MAX, то MAX будет равен этому элементу.

QB TP

Найти максимальный (минимальный) элемента массива, а также его порядковый номер.

Одновременно запоминаем индекс промежуточного максимума.

```
IF A(I)>=MAX THEN                if a[i]>= max then
MAX=A(I):IMAX=I                  begin max:=a[i]; imax:=i; end;
NEXT                              end;
```

Печатаем минимальный элемент и его индекс

```
PRINT MIN; ” номер “; IMIN    writeln(min,'  номер ',imin);
```

Печатаем максимальный элемент и его индекс.

```
PRINT MAX; ”  номер  “; IMAX    writeln(max,' номер ',imax);
                                readkey;
END                              end.
```

Домашнее задание

В предыдущей задаче найдите минимальный элемент среди отрицательных чисел и максимальный элемент среди неотрицательных.

Урок 9

Тема урока: Одномерные массивы. Работа с элементами

План урока:

Проверка домашнего задания.

Решение задач.

Домашнее задание.

Ход урока.

1. В предыдущей задаче найдите минимальный элемент среди отрицательных чисел и максимальный элемент среди неотрицательных.

В условии на нахождение промежуточного минимума добавим условие **A(I)<0**

IF A(I)<=MIN AND A(I)<0 THEN

MIN=A(I):IMIN=I

if (a[i]<=min) and (a[i]<0 then

begin min:=a[i]; imin:=i; end;

В условии на нахождение промежуточного максимума добавим условие **A(I)>=0**

IF A(I)>=MAX AND A(I)>=0 THEN

MAX=A(I):IMAX=I

if (a[i]>= max) and (a[i]>=0) then

begin max:=a[i]; imax:=i; end;

В предыдущей задаче найдите максимальный элемент среди отрицательных чисел и минимальный элемент среди неотрицательных.

Достаточно ли изменить условие в строках?

```
IF A(I)<=MIN AND A(I) >=0 THEN
MIN=A(I):IMIN=I
```

```
if (a[i]<=min) and (a[i]>=0) then
begin min:=a[i];
imin:=i;
end;
```

```
IF A(I)>=MAX AND A(I) < 0 THEN
MAX=A(I):IMAX=I
```

```
if (a[i]>= max) and (a[i]<0) then
begin
max:=a[i]; imax:=i;
end;
```

Ответ: нет, так как в этом случае мы неправильно задали начальный максимум и минимум. В случае, если первый элемент окажется положительным, то мы неправильно найдем максимум среди отрицательных элементов. Если первый элемент окажется отрицательным, то мы неправильно найдем минимум среди неотрицательных элементов.

Подсчитать количество четных элементов массива, заданного датчиком случайных чисел.

```
Prim19

DEFINT I, K, N,R
DIM A(I000) AS INTEGER

CLS
RANDOMIZE TIMER
INPUT “число элементов  N=”;N

FOR I=1 TO N
A(I)=RND*45-22
PRINT A(I);”  “;
```

Проверяем на четность и считаем количество четных элементов.

```
IF A(I) MOD 2 =0 THEN K=K+1
NEXT
PRINT “K=”;K

END
```

```
program Prim19;
Uses crt;
Var i,k,n,r:integer;
a:array[1..10000] of integer;
Begin
clrscr;
randomize;
write(' число элемен.n=');
readln(n);
 for i:=1 to n do     begin
 a[i]:=random(45)-22;
 write(a[i],' ');
 If a[i] mod 2=0 then
 inc(k,1);
 End;
 Write('к= ',k);
 Readkey
end.
```

QB TP

Подсчитать количество отрицательных и нбеотрицательных элементов массива, заданного датчиком случайных чисел

Prim2	program prim2;
	uses crt;
DEF A(1000) AS INTEGER	var a:array [1..1000] of integer;
DEFINT I,K-L,N	i,k,n,l:integer;
	begin
RANDOMIZE TIMER	randomize;
CLS	clrscr;
INPUT “ число элементов N=”;N	write('число элементов n=');readln(n);
FOR I=1 TO N	for i:=1 to n do begin
A(I)=RND*45-22	a[i]:=random(45)-22;
PRINT A(I);” “;	write(a[i],' ');

Сравниваем с нулем и считаем неотрицательные и отрицательные элементы

```
IF A(I)>=0 THEN K=K+1
ELSE L=L+1
NEXT
PRINT
PRINT “неотриц.  =”;K,”
отриц. =”;L

END
```

```
if a[i]>=0 then k:=K+1
else L:=l+1;
end;
writeln;
writeln('неотриц.= ',k,'
отриц. =',l);
 readln;
end.
```

QB TP

Произвести удаление элемента из одномерного массива, заданного датчиком случайных чисел.

После заполнения массива вводим номер удаляемого элемента.

INPUT” номер удаляемого элем. k=<n'); К=< N “;K	**write(' номер удал. элем. readln(k);**
FOR I=1 TO N-1	**for i:=1 to n-1 do begin**

Смещаем элементы массива к началу, начиная с К-го.

IF I>=K THEN A(I)=A(I+1)	**f i>=k then a[i]:=a[i+1];**
NEXT	**end;**

Обнуляем последний элемент

A(N)=0	**a[n]:=0;**

Выводим на экран измененный массив

FOR I=1 TO N-1	**for i:=1 to n -1 do begin**
PRINT A(I);” “;	**write(a[i],' ');**
NEXT	**end;**
	readln;
END	**end.**

QB TP

Произвести вставку элемента в одномерный массив, заданный датчиком случайных чисел.

После заполнения массива вводим индекс и значение вставляемого элемента.

INPUT “ ИНДЕКС <=N “;K	write(' индекс <=N'); readln(k)
INPUT” значение R =”;R	write(' значение R = '); readln(r);

Смещаем элементы к концу массива, начиная с К+1

FOR I=N+1 TO 1 STEP-1	for i:=n+1 downto 1 do begin

QB TP

IF I>K THEN A(I)=A(I-1)	if i>k then a[i]:=a[i-1];
NEXT	end;

На какое место ставим введенный элемент?

A(K)=R	a[k]:=r;

Выводим на экран измененный массив.

FOR I=1 TO N+1	for i:=1 to n +1 do begin
PRINT A(I);” “;	write(a[i],' ');
NEXT	end;
	readln;
END	end.

Домашнее задание

Что необходимо изменить в программе подсчета количества элементов, чтобы в К находилась сумма неотрицательных элементов, а в L произведение отрицательных элементов.

Урок 10

Тема урока: Двумерные массивы. Работа с элементами

План урока:
Проверка домашнего задания.
Двумерные массивы.
Решение задач.
Домашнее задание.

Что необходимо изменить в программе подсчета количества элементов, чтобы в К находилась сумма неотрицательных элементов, а в L произведение отрицательных элементов?

Для этого изменим строку, предварительно задав начальные значения К=0 и L=1.

IF A(I)>=0 THEN K=K+ A(I)	**if a[i]>=0 then k:= k + a[i]**
ELSE L=L* A(I)	**else L:=l+ a[i];**

Двумерные массивы

Массивы, положение элементов в которых описывается двумя индексами, называются **двумерными**.

Их можно представить в виде прямоугольной таблицы или матрицы.

Рассмотрим матрицу А размером 2*3, то есть в ней будет две строки, а в каждой строке по три элемента:

A= a11 a12 a13

a21 a22 a23

a11	*a12*	*a13*
a21	*a22*	*a23*

- Каждый элемент имеет свой номер, как у одномерных массивов, но сейчас номер уже состоит из двух чисел — номера строки, в которой находится элемент, и номера столбца.
- Таким образом, номер элемента определяется пересечением строки и столбца.
- Например, ***a12*** — это элемент, стоящий в первой строке и во втором столбце.

Способы объявления двумерного массива

Способ 1

- В Turbo Pascal 7.0 двумерный массив можно описать как одномерный, элементами которого являются одномерные массивы. Например, для матрицы А, приведенной выше:
- **Const n =2; т=3;**
- **Type. Mas1=Array[1..m] Of <тип элементов >; Mas2=Array[1..n] Of mas1;**
- **Var v: mas1;**
- **a: mas2;**

- В данном случае переменная v объявлена, как одномерный массив из трех элементов вещественного типа. Переменная а описана как двумерный массив из двух строк, в каждую из которых включено по три элемента.

Способ 2

- Описание массива А можно сократить, исключив определение типа mas1 в определении типа mas2:

Const n=2; m=3;

Type massiv = Array[1..n] Of Array[1..m] Of <тип элементов>;

Var a: massiv.

Способ 3

- Еще более краткое описание массива А можно получить, указывая имя массива и диапазоны изменения индексов для каждой размерности массива:

Const п=2; т=3;

Type massiv = Array[1..n,1..mJ Of <mun элементов>;

Var a: massiv.

- Если указанный тип используется для определения одного массива в программе, то удобно объявление массива в разделе описания переменных:

Var a: Array[1..n,1..m] Of <тип элементов>.

На языке QBasic двумерные массивы определяются аналогично, как и одномерные.

Например:

DIM имя массива (n_1 TO n2,n3 TO n4) AS тип элементов

DIM A(1 TO 100, 1 TO 100) AS INTEGER

DIM A(100, 100) AS INTEGER

если n1 =1, то номер можно не указывать.

- Допускается в описании объединять массивы в список, например:

DIM A(50,40), B$(15,50), C%(50,11)

- Здесь объявлено списком три массива A, B, C действительные, строковые и целые.

- Рассмотренные выше методы решения задач обработки одномерных массивов могут применяться для обработки двумерных массивов. Поскольку положение элемента в двумерном массиве описывается двумя индексами (первый — номер строки, второй — номер столбца), программы большинства матричных задач строятся на основе вложенных циклов. Обычно внешний цикл работает по строкам матрицы, то есть с его помощью выбирается требуемая строка матрицы, а внутренний цикл — по столбцам матрицы, то есть здесь выбирается нужный элемент из выбранной уже строки.
- Для задания значений элементам массива могут быть использованы операторы присваивания и операторы ввода данных.
- В приведенном ниже примере осуществляется ввод и вывод двумерного массива A, в котором N строк и N столбцов.
- Формирование двумерного массива можно осуществлять всеми тремя способами, описанными для одномерных массивов, то есть ввод с клавиатуры, через генератор случайных чисел или с помощью файла. Пусть в нашем примере элементы задаются генератором случайных чисел.

Найти сумму всех элементов двумерного массива и сумму элементов каждой строки.

```
‘prim1

DIM A(1 TO 100,1 TO 100)
 AS INTEGER

DEFINT I-J, N,S

CLS
RANDOMIZE TIMER
INPUT “N=”;N
FOR I=1 TO N
FOR J=1 TO N
```

```
program prim1;
uses crt;
Type mas=array
[1..100,1..100] of integer;
var a:mas;
i,j,n,s,s1:integer;
begin
clrscr;
randomize;
write('n=');readln(n);
for i:=1 to n do begin
for j:=1 to n do begin
```

Получаем случайные значения элементов матрицы.

A(I,J)=RND*45-22	a[i,j]:=random(45)-22;
PRINT USING” ####”;A(I,J);	write(a[i,j]:4);

Находим сумму элементов.

S=S+A(I.J): S1=S1+A(I.J)	s:=s+a[i,j]; s1:=s1+a[i,j];
NEXT	end;

Печатаем сумму всех элементов каждой строки и обнуляем значение суммы.

PRINT” сумма ‘элем	writeln(‘сумма элем
строки =”;S:S=0	строки=’,s); s:=0;
PRINT	writeln;
NEXT	end;
PRINT” Сумма всех элем. S1=”;S1	writeln(‘сумма всех элем. s1=’,s1);
	readln;
END	end.

QB TP

Найти максимальный элемент каждой строки массива и его индексы (всего массива и его индексы)

Решение.

Так как элементы могут повторяться, то договоримся, что будем запоминать только индексы первого максимального элемента.

```
prim2

DIM A(1 TO 100,1 TO 100)
AS INTEGER


DEFINT I-J
DEFINT M-N

CLS
RANDOMIZE TIMER
INPUT "N=";N
FOR I=1 TO N
```

```
program prim2;
uses crt;

Type mas=array
[1..100,1..100] of integer;
var a:mas;
i, j, n: integer;
max,min,i1,j1,i2,j2:integer;
begin
clrscr;
 randomize;
 write('n=');readln(n);
for i:=1 to n do begin
```

Продолжение программы

Так как тип массива integer, то за начальные значения возьмем:

MAX = -32768	**max := -32768;**
FOR J=1 TO N	**for j:=1 to n do begin**

Получаем случайные значения элементов матрицы.

A(I,J)=RND*45-22	**a[i,j]:=random(45)-22;**

Выводим элементы матрицы на экран.

PRINT USING” ####”;A(I,J);	**write(a[i,j]:4);**

Находим максимальный элемент в каждой строке и его индексы.

IF A(I,J)>MAX THEN	**if a[i,j]>max then**
	begin
MAX=A(I,J):I1=I:J1=J	**max:=a[i,j];i1:=i;j1:=j;**
	end;
NEXT	**end;**

Печатаем максимальный элемент в каждой строке и его индекс.

PRINT “MAX=”;MAX,” строка =”;I1, “ столбец =”;J1

write(' max=',max, ' строка=',i1, ‘столбец =', j1);

Переходим на следующую строку.

PRINT

NEXT

END

writeln;

end;

readln;

end.

Что необходимо изменить в программе, чтобы находить максимальный элемент во всем массиве.

Для нахождения максимального элемента всего массива необходимо:

- перенести начальный максимум на одну строку выше
- перенести печать максимального элемента на две строки вниз.

QB TP

Найти количество элементов, больших некоторого заданного числа X в каждой строке массива (во всем массиве).

```
`prim25
DIM A(1 TO 100,1 TO 100) AS INTEGER
DEFINT I-J,N
DEFINT K-L,X

CLS
RANDOMIZE TIMER
INPUT"N=";N
```

Задаем значение X.

```
X=0
FOR I=1 TO N
K=0:L=0
FOR J=1 TO N
A(I,J)= RND*45-22
PRINT USING" ####";A(I,J);
```

```
program prim25;
uses crt;
var a:
=array[1..100,1..100] of integer;
i,j,n:integer;
k,l,x:integer;
begin
clrscr;
randomize;
write('n=');readln(n);
x:=0;
for i:=1 to n do begin
k:=0;l:=0;
for j:=1 to n do begin
a[i,j]:=random(45)-22;
write(a[i,j]:4);
```

Продолжение программы

Считаем число элементов, удовлетворяющих условию задачи.

```
IF A(I,J)>X THEN K=K+1          if a[i,j]>x then k:=k+1
ELSE L=L+1                       else l:=l+1;
NEXT                              end;
```

Если находим для всего массива, то следующую строку надо убрать, а начальные значения K=0:L=0 перенести выше на одну строку.

```
PRINT "K=";K,"L=";L:K=0:L=0     write('  k=',k, ' l=',l);
                                k:=0;l:=0;
PRINT                            writeln;
NEXT                            end;
```

Печатаем число элементов, удовлетворяющих условию задачи во всем массиве.

```
PRINT "K=";K,"L=";L              write('  k=',k, ' l=',l);
                                 readln;
END                             end.
```

Домашнее задание.

Измените программу примера так, чтобы находился последний максимальный элемент и его индексы во всем массиве.

QB TP

Урок 11

Тема урока: Языки программирования Qbasic и Turbo Pascal 7.0. Двумерные массивы. Работа с элементами

План урока:
Проверка домашнего задания.
Решение задач.
Домашнее задание.

Проверка домашнего задания.

Измените программу предыдущего примера, чтобы находился последний максимальный элемент и его индексы во всем массиве.

Для решения этой задачи необходимо изменить знак > на => в следующей строке.

IF A(I,J)>=MAX THEN MAX=A(I,J):I1=I:J1=J

if a[i, j]>= max then begin max: = a [i, j]; i1:=i;j1:=j;end;

Продолжим изучение двумерных массивов на конкретных примерах.

Определить, является ли данный квадратный массив симметричным относительно своей главной диагонали.

Если массив является симметричным, то для него выполняется равенство a[i,j]=a[j,i] для всех i=l,..., n и j=1,..., n при условии, что i>j. Но если встретится хотя бы одна такая пара, что соответствующие элементы не будут равны, то массив будет несимметричным.

prim1	**program prim1;**
	uses crt;
DIM A(1 TO 100,1 TO 100) AS INTEGER	
	var a:array[1..100,1..100] of integer;
DEFINT I,J,,N	**i,j,n:integer;**
	begin
CLS	**clrscr;**
RANDOMIZE TIMER	**randomize;**

Продолжение программы

```
INPUT”N=”;N
FOR I=1 TO N
FOR J=1 TO N
A(I,J)=RND*45-22
PRINT USING” ####”;A(I,J);
NEXT
PRINT
NEXT
```

```
write('n=');readln(n);
 for i:=1 to n do begin
 for j:=1 to n do begin
 a[i,j]:=random(45)-22;
 write(a[i,j]:4);
 end;
 writeln;
 end;
 readln;
```

```
FOR I=1 TO N                          for i:=1 to n do
FOR J=1 TO N                          for j:=1 to n do
IF I>J AND A(I,J)<>A(J,I)             (if i>j) and (a[i,j]<>a[j,i])
 THEN                                 then
PRINT ”NO”:
END                                   begin
                                      writeln('no'); exit; end;
NEXT J,I :  PRINT “YES”               writeln('yes');
                                      readln;
END                                   end.
```

Можно ли сократить перебор? Да, так как I>J, то заменим следующие строки:

```
FOR J=1 TO I-1                        for j:=1 to i-1 do
IF A(I,J)<>A(J,I) THEN               (if i>j) and (a[i,j]<>a[j,i])
                                          then
```

QB TP

Вставить строку из нулей на место строки с номером k.

Для решения этой задачи необходимо:

- Первые k-1 строк оставить без изменения.
- Все строки после k-й сдвинуть на одну назад, это лучше начать с последней строки и идти до k-й.
- Элементам строки k присвоить заданное значение.
- Кроме того, необходимо изменить размер массива. Так как мы вставляем строку, то число строк будет на одну больше:

prim2	program prim2;
	uses crt;
DIM A(1 TO 100,1 TO 100) AS INTEGER	Type mas=array[1..100,1..100] of integer;
	var a:mas;
DEFINT I-K,N	i,j,k,n:integer;
	begin
CLS	clrscr;
RANDOMIZE TIMER	randomize;
INPUT”N=”;N	write('n=');readln(n);

Создаем и распечатываем двумерный массив.

FOR I=1 TO N	for i:=1 to n do begin
FOR J=1 TO N	for j:=1 to n do begin
A(I,J)=RND*45-22	a[i,j]:=random(45)-22;

PRINT USING” ####”;A(I,J);	**write(a[i,j]:4);**
NEXT	**end;**
PRINT	**writeln;**
NEXT	**end;**

Вводим номер вставляемой строки.

INPUT”K=”;K	**write('k=');readln(k);**

Сдвигаем строки назад до К-ой строки, начиная с последней.

FOR I= N TO K STEP-1	**for i:=n downto k do**
FOR J=1 TO N:A(I+1,J)=A(I,J):NEXTJ,I	**for j:=1 to n do a[i+1,j]:=a[i,j];**
END	**end.**

Обнуляем элементы К-ой строки.

```
FOR J=1 TO N:A(K,J)=0:NEXT        for j:=1 to n do   a[k,j]:=0;
```

Распечатываем новый массив, который на одну строку больше исходного.

```
FOR I=1 TO N +1                   for i:=1 to n+1 do begin
FOR J=1 TO N                      for j:=1 to n do begin
PRINT USING” ####”;A(I,J);        write(a[i,j]:4);
NEXT                              end;
PRINT                              writeln;
NEXT                              end;
                                   readln;
```

QB TP

Удалить строку с номером k.

Для того, чтобы удалить строку с номером k, необходимо:

- Сдвинуть все строки, начиная с данной, на одну вверх.
- Последнюю строку "обнулить", то есть всем элементам последней строки присвоить нулевое значение.

Будем выводить на экран сначала все строки, а второй раз, после удаления, на одну меньше.

Prim3	**program prim3;**
	uses crt;
DIM A(1 TO 100,1 TO 100)	**Type mas=array [1..100,1..100]**
AS INTEGER	**of integer;**
	var a:mas;
DEFINT I-K,N	**i, j, k, n: integer;**
	begin
CLS	**clrscr;**
RANDOMIZE TIMER	**randomize;**
INPUT”N=”;N	**write('n=');readln(n);**

Создаем и распечатываем двумерный массив.

FOR I= 1 TO N	**for i:=1 to n do begin**
FOR J= 1 TO N	**for j:=1 to n do begin**
A(I,J)=RND*45-22	**a[i,j]:=random(45)-22;**

```
PRINT USING” ####”;A(I,J);        write(a[i,j]:4);
NEXT                              end;
PRINT                             writeln;
NEXT                              end;
```

Вводим номер удаляемой строки

```
INPUT”K=”;K                       write('k=');readln(k);
```

Сдвигаем строки на одну вверх, начиная с данной

```
FOR K=1 TO N-1                    for i:=K to n-1 do
FOR J=1 TO N:A(I,J)=A(I+1,J):     for j:=1 to n do
NEXTJ,I                           a[i,j]:=a[i+1,j];
```

Обнуляем последнюю строку

```
FOR J=1 TO N:A(N,J)=0:NEXT
```

```
for j:=1 to n do   a[n,j]:=0;
```

Печатаем новый массив, в котором на одну строку меньше

```
FOR I=1 TO N-1
FOR J=1 TO N
PRINT USING” ####”;A(I,J);
NEXT
PRINT
NEXT

END
```

```
for i:=1 to n-1 do begin
for j:=1 to n do begin
write(a[i,j]:4);
end;
writeln;
end;
readln;
end.
```

QB TP

Поменять местами строки с номерами К1 и К2.

prim4	program prim4;
	uses crt;
	Type mas=array [1..100,1..100]
	of integer;
DIM A(1 TO 100,1 TO 100)	var a:mas;
AS INTEGER	
DEFINT I-K,N,R	i,j,n,k1,k2,r:integer;
	begin
CLS	clrscr;
RANDOMIZE TIMER	randomize;
INPUT”N=”;N	write('n='); readln(n);

Создаем и распечатываем двумерный массив

```
FOR I= 1 TO N
FOR J= 1 TO N
A(I,J)=RND*45-22
PRINT USING” ####”;A(I,J);
NEXT
PRINT
NEXT
```

```
for i:=1 to n do begin
for j:=1 to n do begin
a[i,j]:=random(45)-22;
write(a[i,j]:4);
end;
writeln;
end;
```

Продолжение программы

Вводим номера строк, которые будем менять местами

INPUT”K1=,K2=”;K1,K2	**write('stroki k1=k2=');** **readln(k1,k2);**

Меняем значения К1 и К2 строк между собой.

FOR J=1 TO N	**for j:=1 to n do begin**
R=A(K1,J):A(K1,J)=A(K2,J):A(K2,J)=R	
	r:= a[k1,j];a[k1,j]:=a[k2,j]; **a[k2,j]:=r;**
NEXT	**end;**

QB TP

Распечатаем измененный массив

```
FOR I=1 TO N
FOR J=1 TO N
PRINT USING” ####”;A(I,J);
NEXT
PRINT
NEXT

END
```

```
for i:=1 to n do begin
for j:=1 to n do begin
 write(a[i,j]:4);
 end;
 writeln;
 end;
 readln;
end.
```

Домашнее задание.

В массиве размером п*п к элементам четных столбцов прибавить элемент первого столбца соответствующей строки.

Урок 12

Тема урока: Языки программирования QBasic и Turbo Pascal 7.0. Двумерные массивы. Работа с элементами

План урока:
Проверка домашнего задания.
Решение задач.
Домашнее задание.

В массиве размером n n к элементам четных столбцов прибавить элемент первого столбца соответствующей строки.*

Пропустим начальные строки программы. В данной задаче необходимо проверять четность столбцов. Основная часть программы будет выглядеть так.

FOR I= 1 TO N	**for i:=1 to n do begin**
FOR J=1 TO N	**for j:=1 to n do**

Проверяем номер столбца на четность. В случае четности прибавляем к значениям проверяемого столбца соответствующие значения первого столбца.

IF J MOD 2=0 THEN	**if j mod 2=0 then**
A(I,J)=A(I,J)+A(I,1)	**a[i,j]:=a[i,j]+a[i,1];**
NEXT J,I	**end;**

Далее распечатываем измененный массив

Иногда массив необходимо заполнять не случайными числами, а по некоторому закону.

Пример 1

Заполнить массив А размером n*m следующим образом, например, **n=5 и m=5:**

1	**2**	**3**	**4**	**5**
10	**9**	**8**	**7**	**6**
11	**12**	**13**	**14**	**15**
16	**17**	**18**	**19**	**20**
21	**22**	**23**	**24**	**25**

То есть заполняется в виде "змейки".

Решение.

Для того, чтобы заполнить, надо вывести правило заполнения, а оно в данном случае будет таким:

если ряд нечетный (то есть номер строки — нечетное число),

то **A[i,j]=(i-l)*m+j,** иначе (то есть когда строка четная) **A[i,j]==i*m-j+l.**

По этому правилу и составляем процедуру заполнения. Тем более, что проверку на четность мы производили (только со столбцами) в домашней работе.

prim1

```
DIM A(1 TO 100,1 TO 100)
AS INTEGER

DEFINT I-N

CLS
INPUT”N=M=”;N,M

FOR I=1 TO N
FOR J=1 TO M
```

```
program prim1;
uses crt;
 Type mas=array
[1..100,1..100] of integer;

var a:mas;
i,j,n, m:integer;
begin
clrscr;

write('n=m=');readln(n,m);
for i:=1 to n do begin
for j:=1 to m do begin
```

Заполняем массив по выведенному правилу заполнения и выводим его на экран.

```
IF I MOD 2 =1 THEN
A(I,J)=(I-1)*M+J
ELSE A(I,J)=I*M-J+1
END IF
PRINT USING” ####”;A(I,J);
NEXT
PRINT
NEXT

END
```

```
if i mod 2 =1 then
a[i,j]:=(i-1)*m+j
else a[i,j]:=i*m-j+1;

write(a[i,j]:4);
end;
writeln;
end;
readln;
end.
```

QB TP

Составить программу, запрашивающую координаты ферзя на шахматной доске и показывающую поля доски, находящиеся под боем.

Решение.

- Шахматную доску представим в виде двумерного массива размером 8*8. Координаты ферзя мы будем вводить двумя числами (номер строки и номер столбца), хотя в шахматах принято вводить букву и число. (Буква отвечает за номер строки, а число — за номер столбца). Это мы делаем для сокращения программы. При желании вы можете не отступать от традиций и вводить координаты именно таким образом. В программе необходимо будет перевести букву в соответствующее ей число ('a' - 1, V - 2, 'c' - 3,'d' - 4, 'e' - 5,'f'—6, 'g'— 7,'h'— 8), тогда будет удобнее работать.

Для решения надо еще знать следующие свойства шахматной доски. Все диагонали делятся на восходящие и нисходящие:

На каждой диагонали выполняется свойство:

- для любой восходящей диагонали сумма номера строки и номера столбца постоянна и для разных диагоналей — разная, то есть i + j = constl;
- для нисходящих — разность номера строки и номера столбца тоже постоянна и для разных диагоналей — разная, то есть i - j = const2.

Это необходимо для того, чтобы определить номера диагоналей, на которых поставлен ферзь. Вся программа будет такой:

prim2	**program prim2;**
	uses crt;
DIM A(1 TO 100,1 TO 100)	**Type mas=array**
AS INTEGER	**[1..100,1..100] of integer;**
	var a:mas;
DEFINT I-L,N	**i,j,n,k,l:integer;**
	begin
CLS	**clrscr;**

INPUT”N=”;N
write('n=');readln(n);

Вводим координаты ферзя K<=N и L<=N.

INPUT”K=,L=”;K,L	**write('k=l=');readln(k,l);**
FOR I= 1 TO N	**for i:=1 to n do begin**
FOR J=1 TO N	**for j:=1 to n do begin**

Для нисходящих и восходящих диагоналей, проходящих через точку (k,l), присваиваем значения единицы, а других нуль.

IF I=K OR J=L OR I+J=K+L OR I-J=K-L THEN A(I,J)=1
ELSE A(I,J)=0

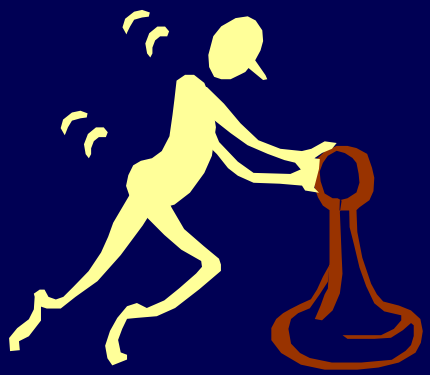
if (i=k) or (j=l) or (i+j=k+L) or (i-j=k-L) then a[i,j]:=1
else a[i,j]:=0;

Клетке, где стоит ферзь, присваиваем значение два.

A(K,L)=2 **a[k,l]:=2;**

Распечатываем массив

PRINT USING” ####”;A(I,J); **write(a[i,j]:4);**
NEXT **end;**
PRINT **writeln;**
NEXT **end;**
readln;
END **end.**



QB TP

Ввести координаты ферзя и коня и определить: бьёт ли ферзь коня, если первый ход ферзя.

Задача аналогична предыдущей, только в ней дополнительно придется проверять, не стоит ли конь под ударом ферзя.

prim3	**program prim3;**
	uses crt;
DIM A(100,100) AS INTEGER	**Type mas=array [1..100,1..100] of integer;**
	var a:mas;
DEFINT I-L,N	**i,j,n,k,l,k1,l1:integer;**
	begin
CLS	**clrscr;**
INPUT”N=”;N	**write('n=');readln(n);**

Вводим координаты ферзя

```
INPUT”K=,L=”;K,L
   write('k=l=');readln(k,l);

FOR I=1 TO N
FOR J=1 TO N
   begin
```

```
for i:=1 to n do begin
   for j:=1 to n do
```

Заполняем поля, которые бьет ферзь единицами,

а где стоит ферзь двойкой

```
IF I=K OR J=L OR I+J=K+L OR I-J=K-L THEN A(I,J)=1 ELSE
   A(I,J)=0
A(I,J)=2
```

```
if ( i=k) or (j=l) or (i+j=k+L) or (i-j=k-L)  then   a[i,j]:=1 else
   a[i,j]:=0;
a[k,l]:=2;
```

Распечатываем массив.

PRINT USING” ####”;A(I,J);	**write(a[i,j]:4);**
NEXT	**end;**
PRINT	**writeln;**
NEXT	**end;**

Вводим координаты коня

INPUT”КООРДИНАТЫ КОНЯ=”;	**write('koordin kon**
K1,L1	**k1=l1=');readln(k1,l1);**

Проверяем, бьется ли данное поле ферзем

IF A(K1,L1)=1 THEN PRINT”БЬЕТ”	**if a[k1,l1]=1 then writeln ('бьет')**
ELSE PRINT ”НЕТ”	**else writeln('нет');**
	readln;
END	**end.**

Домашнее задание.

Подготовиться к контрольной работе.

QB TP

Контрольная работа № 3

Вариант № 1

1. Дан массив целых чисел, состоящий из 20 элементов.

Заполнить его с клавиатуры. Найти:

- сумму элементов, имеющих нечетное значение;
- вывести индексы тех элементов, значения которых больше заданного числа А.

2. Определить, есть ли в данном массиве положительные элементы, кратные k (k вводить с клавиатуры).

Вариант № 2

1. Дан массив целых чисел, состоящий из 25 элементов. Заполнить его с клавиатуры. Найти:

- сумму элементов, имеющих нечетные индексы;
- подсчитать количество элементов массива, значения которых больше заданного числа А и кратны 5.

2. Найти номер первого отрицательного элемента, делящегося на 5 с остатком 2.

Вариант № 3

1. Дан массив целых чисел, состоящий из 15 элементов. Заполнить его с клавиатуры. Найти:

- сумму положительных элементов, значения которых меньше 10;

- вывести индексы тех элементов, значения которых кратны 3 и 5.

2. Определить, есть ли пара соседних элементов с суммой, равной заданному числу.

Вариант № 4

1. Дан массив целых чисел, состоящий из 10 элементов. Заполнить его с клавиатуры. Найти:

- удвоенную сумму положительных элементов;

- вывести индексы тех элементов, значения которых больше значения предыдущего элемента (начиная со второго).

2. Определить, есть ли 2 пары соседних элементов с одинаковыми знаками.

Вариант № 5

1. Дан массив целых чисел, состоящий из 30 элементов. Заполнить его с клавиатуры. Найти:

- сумму отрицательных элементов;
- найти количество тех элементов, значения которых положительны и не превосходят заданного числа А.

2. Найти номер последней пары соседних элементов с разными знаками.

Вариант № 6

1. Заменить максимальный по модулю отрицательный элемент нулем.
2. Заменить первые k элементов на противоположные по знаку.
3. Из элементов массива С сформировать массив А того же размера по правилу: если номер четный, то Ai=C \2, если нечетный, то Ai=2*Ci.

Вариант № 7

1. Заменить минимальный по модулю положительный элемент нулем.

2. Заменить элементы с k1-го по k2-й на обратные.

3. Из элементов массива A сформировать массив D того же размера по правилу: первые 10 элементов — Di=Ai+i, остальные — Di=Ai—i.

Вариант № 8

1. Заменить первый отрицательный элемент нулем.

2. Умножить все элементы, кратные 3, на третий элемент массива.

3. Из элементов массива P сформировать массив M того же размера по правилу: если номер четный, то Pi=i*Mi, если нечетный, то Pi=-Mi.

Вариант № 9

1. Заменить максимальный элемент на противоположный по знаку.

2. Заменить нулями элементы между минимальным и максимальным, кроме них самих.

3. Из элементов массива C сформировать массив A того же размера по правилу: элементы с 3-го по 12-й — Ai=—Ci2, все остальные — Ai=Ci-l.

Вариант № 10

1. Заменить первый элемент, кратный 5, нулем.

2. Заменить элементы с нечетными номерами на квадрат их номера.

3. Из элементов массива D сформировать массив A того же размера по правилу: если номер четный, то Ai=Di2, если нечетный, то Ai=Di/i.

Вариант № 11

1. Заменить последний положительный элемент на второй элемент массива.

2. Разделить все элементы с четными номерами на первый элемент.

3. Из элементов массива С сформировать массив А того же размера по правилу: если номер четный, то Ai=C\2, если нечетный, то Ai=2Ci**.**

Вариант № 12

1. Дан массив целых чисел (n=15), заполненный случайным образом числами из промежутка [—20,50].

- удалить из него все элементы, в которых есть цифра 5.

- вставить число k после всех элементов, кратных своему номеру (k - с клавиатуры).

- поменять местами первый положительный и последний отрицательный элементы.

Вариант № 13

1. Дан массив целых чисел (n=10), заполненный случайным образом числами из промежутка [—40,30].

- удалить из него все элементы, которые состоят из одинаковых цифр (включая однозначные числа).
- вставить число k перед всеми элементами, в которых есть цифра 1 (k вводить с клавиатуры).
- переставить первые три и последние три элемента местами, сохраняя их следование.

Вариант № 14

1. Дан массив целых чисел (n=12), заполненный случайным образом числами из промежутка [—10,60].

- удалить из него все элементы, в которых последняя цифра четная, а само число делится на нее.
- вставить число k перед и после всех элементов, заканчивающихся на данную цифру (k вводить с клавиатуры).
- переставить элементы следующим образом: a[1], a[12], a[2], a[11], ..., a[5], a[8], a[6], a[7].

Контрольная работа № 3

Вариант № 15

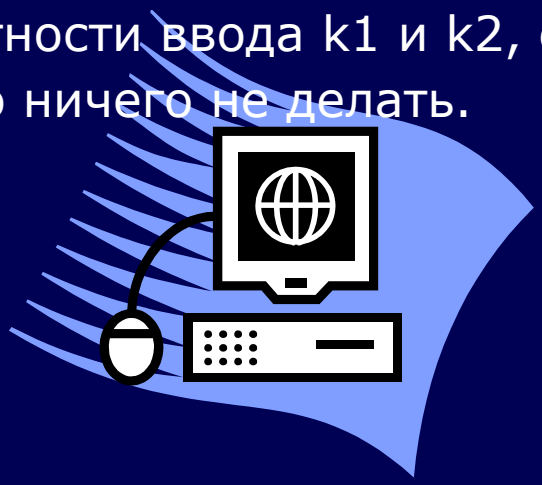
1. Дан массив целых чисел (n=25), заполненный случайным образом числами из промежутка [—35,75].

- удалить из него все элементы, первая цифра которых четная.
- вставить число k1 после всех элементов, больших заданного числа, а число k2 — перед всеми элементами, кратными 3 (k1 и k2 вводить с клавиатуры).
- перенести первые k элементов в конец, то есть: a[k+l], a[k+2],...,a[n], a[l], a[2], ... a[k].

Вариант № 16

1. Дан массив целых чисел (n=20), заполненный случайным образом числами из промежутка [-45,95].

- удалить из него все элементы, кратные 7 и принадлежащие промежутку [a, n] (a и b вводить с клавиатуры).
- вставить число k между всеми соседними элементами, которые образуют пару элементов с одинаковыми знаками (k вводить с клавиатуры).
- переставить в обратном порядке часть массива между элементами с номерами k1 и k2, включая их. Сделать проверку корректности ввода k1 и k2, если ввод неправильный, то ничего не делать.



Двумерные массивы. Контрольная работа № 4

Вариант № 1

1. Дан двумерный массив размером 5*6, заполненный целыми числами с клавиатуры. Сформировать одномерный массив, каждый элемент которого равен произведению четных положительных элементов соответствующего столбца.

2. Дан двумерный массив размером n*m, заполненный случайным образом. Определить, есть ли в данном массиве строка, в которой ровно два отрицательных элемента.

3. Заполнить массив размером 7*7 по правилу:

1 0 0 0 0 0 1
0 1 0 0 0 1 0
0 0 1 0 1 0 0
0 0 0 1 0 0 0
0 0 1 0 1 0 0
0 1 0 0 0 1 0
1 0 0 0 0 0 1

Вариант № 2

1. Дан двумерный массив размером 4*6, заполненный целыми числами с клавиатуры. Сформировать одномерный массив, каждый элемент которого равен количеству элементов соответствующей строки, больших данного числа.

2. Дан двумерный массив размером n*m, заполненный случайными числами. Определить, есть ли в данном массиве столбец, в котором имеются одинаковые элементы.

3. Заполнить массив размером 7*7 по правилу:

```
1 1 1 1 1 1 1
0 1 1 1 1 1 0
0 0 1 1 1 0 0
0 0 0 1 0 0 0
0 0 1 1 1 0 0
0 1 1 1 1 1 0
1 1 1 1 1 1 1
```

Вариант № 3

1. Дан двумерный массив размером 5*6, заполненный целыми числами с клавиатуры. Сформировать одномерный массив, каждый элемент которого равен наибольшему по модулю элементу соответствующего столбца.

2. Дан двумерный массив размером п*т, заполненный случайными числами. Определить, есть ли в данном массиве строка, в которой имеется два максимальных элемента всего массива.

3. Заполнить массив размером 6*6 по правилу:

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

Вариант № 4

1. Дан двумерный массив размером 4*5, заполненный целыми числами с клавиатуры. Сформировать одномерный массив, каждый элемент которого равен количеству отрицательных элементов, кратных 3 или 5, соответствующей строки.

2. Дан двумерный массив размером п*т, заполненный случайными числами. Определить, есть ли в данном массиве столбец, в котором равное количество положительных и отрицательных элементов.

3. Заполнить массив размером 6*6 по правилу**:**

1	**1**	**1**	**1**	**1**	**1**
1	**2**	**3**	**4**	**5**	**6**
1	**3**	**6**	**10**	**15**	**21**
1	**4**	**10**	**20**	**35**	**56**
1	**5**	**15**	**35**	**70**	**126**
1	**6**	**21**	**56**	**126**	**252**

Вариант № 5

1. Дан двумерный массив размером 6*5, заполненный целыми числами с клавиатуры. Сформировать одномерный массив, каждый элемент которого равен первому четному элементу соответствующего столбца, если такого нет, то равен нулю.

2. Дан двумерный массив размером п*т, заполненный случайными числами. Определить, есть ли в данном массиве строка, содержащая больше положительных элементов, чем отрицательных.

3. Заполнить массив размером 7*7 по правилу**:**

1 0 0 1 0 0 1
0 1 0 1 0 1 0
0 0 1 1 1 0 0
1 1 1 1 1 1 1
0 0 1 1 1 0 0
0 1 0 1 0 1 0
1 0 0 1 0 0 1

Вариант № 6

1. Дан двумерный массив размером 5*6, заполненный случайным образом.

а) заменить максимальный элемент каждой строки на противоположный.

б) вставить после столбцов с максимальными элементами столбец из нулей.

в) удалить среднюю строку.

г) поменять местами средние столбцы.

Вариант № 7

1. Дан двумерный массив размером 8*7, заполненный случайным образом.

а) заменить все элементы первых трех столбцов на их квадраты.

б) вставить между средними строками первую строку.

в) удалить все столбцы, в которых первый элемент больше последнего.

г) поменять местами средние строки с первой и последней.

Вариант № 8

1. Дан двумерный массив размером 5*8, заполненный случайным образом.

а) заменить все симметричные элементы на нули.

б) вставить перед всеми строками, первый элемент которых делится на 3, строку из нулей.

в) удалить столбец, в котором находится первый четный отрицательный элемент.

г) поменять местами средние столбцы со вторым и предпоследним.