

24. В одномерном массиве $A=(a_1, a_2, \dots, a_n)$ группу, содержащую наибольшее число подряд идущих положительных элементов, переписать в «хвост» массива.
25. В одномерном массиве $A=(a_1, a_2, \dots, a_n)$ все положительные элементы, расположенные между отрицательными, поставить после минимального элемента массива.

Лабораторная работа № 6

Обработка двумерных массивов

Задание: составить программу заданной обработки матрицы целых чисел. В процессе обработки использовать перестановки элементов внутри массива, не создавая новых массивов. Заполнение исходного массива организовать с помощью генератора случайных чисел. Исходный и обработанный массив выводить на экран. Во всех вариантах работать с ЗАДАНОЙ матрицей, не создавая дополнительных массивов и матриц (кроме случаев, где это предусмотрено вариантом задания).

Теоретический материал

Матрица, по сути, является двумерным массивом элементов, поэтому для работы с матрицей будем использовать двумерный массив. Правила создания, обработки двумерных массивов, в общем, остаются такими же, как и для одномерных массивов. Отличие состоит в количестве индексов.

Объявление двумерного массива можно осуществлять двумя способами. Первый предполагает объявление одномерного массива, а затем использование этого типа для объявления типа-массива массивов:

```
Type mas = array[1..10] of integer;  
    matr = array[1..5] of mas;  
Var a: matr;
```

Однако проще сразу объявить тип-массив, описывающий матрицу, состоящую из 5 строк и 10 столбцов:

```
Type matr = array[1..5,1..10] of integer;  
Var a: matr;
```

Доступ к элементам двумерного массива также может осуществляться двумя способами:

`a[[1],[1]]` или гораздо проще `a[1,1]`

В обоих случаях первый индекс указывает номер строки, второй – номер столбца матрицы.

В памяти элементы матрицы располагаются последовательно:

`a[1,1], a[1,2], ..., a[1,10], a[2,1], a[2,2], ..., a[5,10]`

однако последовательный доступ к элементам матрицы может быть организован как в порядке расположения их в памяти, так и иным способом.

Для организации доступа последовательно ко всем элементам матрицы организуются два цикла, причем порядок «вложения» циклов определяется тем, как осуществляется перебор элементов: по столбцам или по строкам. Чаще используется проход по матрице по строкам, т. е. сначала в первой строке перебираются элементы всех столбцов, затем – во второй строке и т. д. В этом случае «внешним» будет цикл по строкам (т. е. по индексу i), а вложенным – цикл по столбцам (т. е. по j). Ниже представлен пример заполнения матрицы значениями, вводимыми с клавиатуры, причем элементы строк матрицы могут вводиться в строчку через пробел, а для ввода значений новой строки надо нажать клавишу <Enter>.

```
for i:=1 to 5 do
begin
  for j:=1 to 10 do
    read(a[i,j]);
  readln
end;
```

Обработка и вывод матрицы осуществляется аналогично.

Пример выполнения лабораторной работы

Для примера рассмотрим следующую задачу. Дана матрица $A(n \times m)$ целых положительных двузначных чисел. Минимальные элементы строк заменить нулями. Исходную матрицу сгенерировать случайным образом.

Для решения данной задачи составим блок-схему алгоритма (рис. 6.1).

Данный алгоритм не содержит детального описания некоторых блоков (информационного, блока формирования исходной матрицы и вывода результата), т.к. эти блоки не представляют большой сложности и поэтому будем сразу описывать их программным кодом.

Программу, как обычно, начнем с описания констант, типов и переменных. Размеры матрицы (количество строк и столбцов) в ходе выполнения программы не будут изменяться, поэтому они объявляются как константы и задаются в разделе описания констант.

```
const n=5;
      m=7;
```

При описании типа эти константы будут использованы для определения диапазонов изменения индексов матрицы.

```
type matr=array[1..n,1..m] of byte;
```

Тип элементов массива выбран `byte` (возможные значения этого типа данных $[0..255]$), т. к. по условию задачи элементы матрицы положительны и двузначны, т. е. находятся в интервале $[10..99]$. Это уточнение пригодится и при заполнении матрицы.

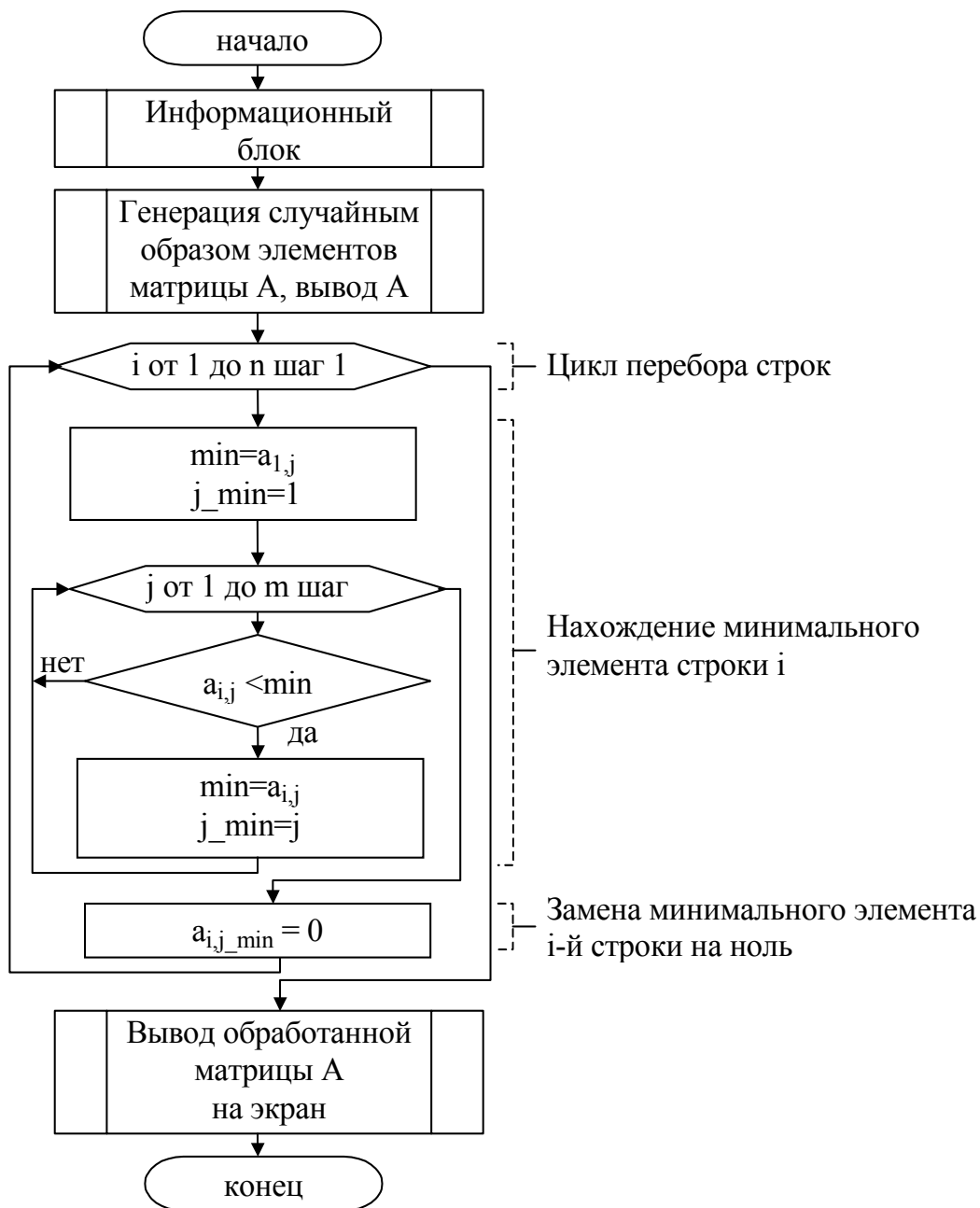


Рис. 6.1. Блок-схема алгоритма замены минимальных элементов строк матрицы A (n x m) на ноль

В разделе описания переменных объявляются переменные для хранения матрицы, индексов, а также минимального значения. Индексы и переменная, которая будет хранить значение минимального элемента, также описываются целочисленным типом `byte`.

```
var a:matr;
    i, j, j_min, min:byte;
```

Далее следует тело программы. Информационный блок подробно рассматриваться не будет. Блок формирования случайным образом элементов матрицы и вывода исходной матрицы выглядит следующим образом:

```
writeln('Исходная матрица');
for i:=1 to n do
begin
  for j:=1 to m do
  begin
    a[i,j]:=random(90)+10;
    write(a[i,j]:3);
  end;
  writeln;
end;
```

До начала цикла по формированию матрицы выводится надпись «Исходная матрица». Далее в цикле формируются элементы сначала первой строки и выводятся на экран в строчку (оператор `write`), затем осуществляется переход на новую строку (`writeln`) и начинается новая итерация цикла по `i`.

Следующий шаг – поиск минимального элемента в каждой строке и замена его на ноль.

```
for i:=1 to n do
begin
  j_min:=1;
  min:=a[i,1];
  for j:=1 to m do
    if a[i,j]<min then
    begin
      min:=a[i,j];
      j_min:=j
    end;
  a[i,j_min]:=0;;
end;
```

Здесь также используется два цикла. Тело «внешнего» цикла содержит операторы присваивания начальных значений переменным `j_min`, `min` и цикл перебора элементов `i`-й строки по столбцам.

Тело «внутреннего» цикла состоит лишь из одного условного оператора, поэтому в нем не нужны операторные скобки. А вот внутри условного оператора – несколько операторов, поэтому операторные скобки `begin..end` в нем **обязательны!**

После завершения цикла по столбцам будет найден минимальный элемент `i`-й строки и его положение (`j_min`), и произведена замена его на ноль. Далее программа возвращается во «внешний» цикл и происходит аналогичная обработка следующей строки матрицы. Выход из внешнего цикла означает, что обработка матрицы завершена. Остается вывести результат на экран, снабдив выведенную матрицу соответствующим текстом.

```
writeln('Результирующая матрица');
for i:=1 to n do
```

```

begin
  for j:=1 to m do
    write(a[i,j]:3);
  writeln;
end;

```

Как можно заметить, этот фрагмент очень похож на блок формирования и вывода исходной матрицы. Отличие состоит в том, что тело «внутреннего» цикла содержит всего один оператор вывода, поэтому он не заключается в операторные скобки begin..end.

Итак, соединив фрагменты в единое целое, получим текст программы:

```

program lab_6;
uses crt;
const n=5;
      m=7;
type matr=array[1..n,1..m] of byte;
var a:matr;
    i,j,j_min,min:byte;
begin
  writeln;
  writeln('      Автор - Иванов И.П., студент гр. ИСЭд-11');
  writeln('      Вариант № 100');
  writeln(' Дана матрица A (n×m) целых положительных двузначных
чисел. ');
  writeln(' Минимальные элементы строк заменить нулями. ');
  writeln(' Исходную матрицу сгенерировать случайным. ');
  writeln;
  {$R+}      {Включение контроля значений индексов}
  clrscr;   {Очистка экрана}
  writeln('Исходная матрица');
  for i:=1 to n do
  begin
    for j:=1 to m do
    begin
      a[i,j]:=random(90)+10;
      write(a[i,j]:3);
    end;
    writeln;
  end;
  for i:=1 to n do
  begin
    j_min:=1;
    min:=a[i,1];
    for j:=1 to m do

```

```

        if a[i,j]<min then
        begin
            min:=a[i,j];
            j_min:=j
        end;
        a[i,j_min]:=0;;
    end;
writeln;
writeln('Результирующая матрица');
for i:=1 to n do
begin
    for j:=1 to m do
        write(a[i,j]:3);
    writeln;
end;
readln;
{$R-}
end.

```

Варианты заданий

1. Даны матрица $A(n \times (n+1))$ и два одномерных массива $X=(x_1, \dots, x_{n+1})$ и $Y=(y_1, \dots, y_{n+1})$, а также натуральные числа p, q . Образовать новую матрицу размера $(n+1) \times (n+2)$ вставкой после строки с номером p матрицы A новой строки с элементами x_1, x_2, \dots, x_{n+1} и последующей вставкой после столбца с номером q нового столбца с элементами y_1, y_2, \dots, y_{n+1} .
2. Даны массив $A=(a_1, a_2, \dots, a_{10})$ и матрица $B(n \times n)$. Заменить нулями в матрице те элементы с четной суммой индексов, для которых имеются равные среди элементов массива A .
3. Даны массив $A=(a_1, a_2, \dots, a_{10})$ и матрица $B(n \times n)$. Элементы первой строки матрицы упорядочены по возрастанию. Получить новую матрицу размера $n \times (n+1)$, вставив в исходную матрицу новый столбец с элементами массива A так, чтобы упорядоченность первой строки матрицы не нарушилась.
4. Дана матрица $A(n \times m)$. Получить матрицу, получающуюся из данной: перестановкой столбцов: первого с последним, второго с предпоследним и т. д.
5. Дана матрица $A(n \times m)$ и целые числа p и q . Преобразовать матрицу A так, чтобы строка с исходным номером p непосредственно следовала за строкой с исходным номером q , сохранив порядок следования остальных строк.
6. Дана матрица A размерности $n \times n$. Найти и вывести ту строку в этой матрице, которая содержит наибольшее количество четных чисел.

7. Дана матрица $A(n \times m)$. Получить матрицу, получающуюся из данной: перестановкой строк - первой с последней, второй с предпоследней и т. д. Назовем допустимым преобразованием матрицы перестановку двух строк и двух столбцов.
8. Дана квадратная матрица порядка n . С помощью допустимых преобразований добиться того, чтобы один из элементов матрицы, обладающий наименьшим по модулю значением, располагался в правом нижнем углу матрицы.
9. Дана матрица $A(n \times m)$. Скорректировать эту матрицу, удалив из нее строки и столбцы, на пересечении которых расположен элемент с наибольшим по модулю значением.
10. Дана матрица $A(n \times n)$, все элементы которой различны. Найти наибольший элемент среди стоящих на главной и побочной диагоналях и поменять его местами с элементом, стоящим на пересечении этих диагоналей.
11. Сформировать массив из отрицательных элементов той строки матрицы A размерности $n \times n$, где обнаружен наименьший элемент этой матрицы.
12. Сформировать массив из положительных элементов той строки матрицы A размерности $n \times n$, где обнаружен наибольший элемент этой матрицы.
13. С помощью матрицы $A(2 \times n)$ в плоскости задано n точек так, что $a_{1,j}$, $a_{2,j}$ – координаты j -й точки. Точки попарно соединены отрезками. Найти длину наибольшего отрезка и вывести его координаты.
14. Дана матрица $A(n \times n)$. Получить $x_1x_n + x_2x_{n-1} + \dots + x_nx_1$, где x_k – наибольшее значение элементов k -го столбца данной матрицы.
15. Дана матрица $A(n \times n)$ многоразрядных целых чисел. Найти номер строки и номер столбца, на пересечении которых находится элемент матрицы, в значении которого больше всего расположено разрядов.
16. Удалить из матрицы $A(m \times n)$ строку, содержащую наибольшее количество нулевых элементов.
17. Дана матрица $A(n \times m)$. Скорректировать эту матрицу, удалив из нее строку и столбец, на пересечении которых расположен элемент с наибольшим по модулю значением.
18. Дана матрица $A(n \times n)$. Сформировать одномерный массив из положительных элементов этой матрицы, расположенных выше главной диагонали.
19. Дана матрица $A(n \times n)$. Сформировать одномерный массив из отрицательных элементов этой матрицы, расположенных ниже главной диагонали.
20. Дана целочисленная матрица $A(n \times n)$. Найти наименьшие элементы в каждой строке этой матрицы и найти количество четных чисел среди них.

21. Дана целочисленная матрица $A(n \times n)$. Найти наибольшие элементы в каждом столбце этой матрице и найти количество нечетных чисел среди них.
22. Дана целочисленная матрица $A(n \times n)$. Элементы тех строк матрицы A , которые начинаются с отрицательного элемента, записать в отдельный одномерный массив и его вывести.
23. Дана целочисленная матрица $A(n \times n)$. Все элементы тех строк, у которых на главной диагонали обнаружен нулевой элемент, заменить на элементы из массива $V=(b_1, \dots, b_n)$.
24. Дана целочисленная матрица $A(n \times n)$. Все элементы тех строк, у которых обнаружено более одного отрицательного элемента, заменить на элементы из массива $V=(b_1, \dots, b_n)$.
25. Дана целочисленная матрица $A(n \times n)$. Все элементы тех столбцов, у которых обнаружено менее 10 положительных элементов, заменить на элементы из массива $V=(b_1, \dots, b_n)$.