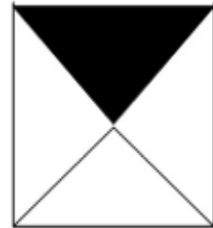


Вариант 12

Вариант 12

Дана матрица.

- а) В каждой строке поместите нулевые элементы в конец строки.
- б) Проверить, равны ли поэлементно i -ая строка и i -ый столбец квадратной матрицы.
- в) Найдите максимальный элемент среди элементов матрицы, выделенных чёрным цветом (матрица квадратная).



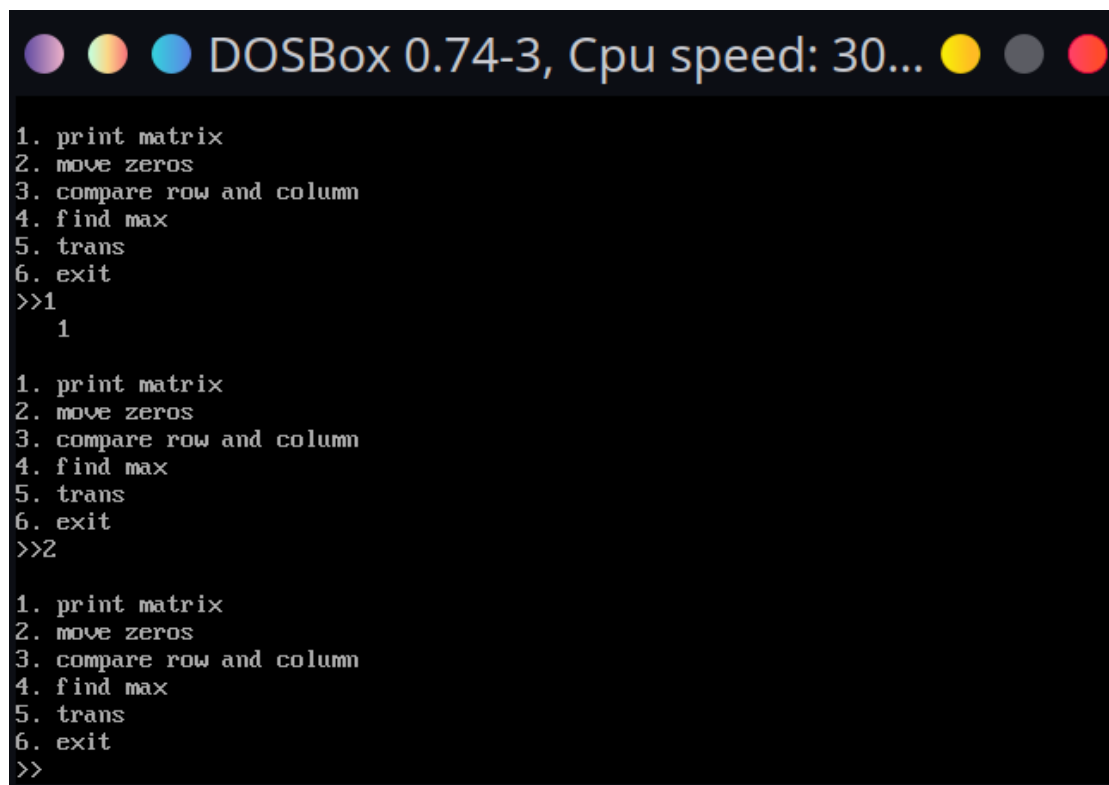
Постановка задачи

Работа предусматривает применение в основных приемах обработки массивов: создание массивов случайным образом с использованием датчика случайных чисел, ввод с клавиатуры, задание массивов по определенному закону, нахождение максимального и минимального элементов массива, перестановка строк и столбцов матрицы, сортировка строк и столбцов, с использованием алгоритма сортировки одномерного массива, перестановка блоков внутри матрицы, умножение матриц.

1. Ввести с клавиатуры и вывести на экран матрицу $m \times n$ (матрица не обязательно должна быть квадратная и может содержать нулевые и отрицательные элементы, если это предусмотрено условиями задания варианта);
2. Реализовать простейший интерфейс взаимодействия с пользователем для выполнения задания варианта до выбора команды «Выход»;
 - Транспонировать матрицу, результат вынести на экран;
 - Обработка элементов матрицы (задание а, б, в условии варианта), результат выполнения отобразить на экране;
 - Реализовать завершение выполнения программы.

Результаты выполнения:

Меню



The image shows a DOSBox window with a title bar that reads "DOSBox 0.74-3, Cpu speed: 30...". The window has three colored window control buttons (purple, yellow, red) on the left and three more (yellow, grey, red) on the right. The main area is a black terminal with white text. It displays a menu with six options: 1. print matrix, 2. move zeros, 3. compare row and column, 4. find max, 5. trans, and 6. exit. The user has entered ">>1" and the program has responded with "1". The menu is repeated twice more, with the user entering ">>2" and ">>" respectively.

```
DOSBox 0.74-3, Cpu speed: 30...
1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>1
1
1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>2
1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>
```

Вывод матрицы после заполнения:

```
DOSBox 0.74-3, Cpu speed: 30...
981
234
21
-21

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>1
-100 -111 -123  -1  -2
   0   1   0   2  -2
   0   1   2  99 -99
   1   2   3   4   5
  991  981  234  21 -21

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>
```

Перемещение нулей в конец:

```
DOSBox 0.74-3, Cpu speed: 30...
4. find max
5. trans
6. exit
>>2

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>1
-100 -111 -123  -1  -2
   1   2  -2   0   0
   1   2  99 -99   0
   1   2   3   4   5
  991  981  234  21 -21

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>
```

Сравнение ряда и колонки по заданному индексу:

```
DOSBox 0.74-3, Cpu speed: 30...
6. exit
>>1
-100 -111 -123 -1 -2
  1   2   -2   0   0
  1   2   99 -99   0
  1   2   3   4   5
991  981 234  21 -21

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>3
Input index>>3
Not equal in second task

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>_
```

Нахождение максимального элемента в выделенной области:

```
DOSBox 0.74-3, Cpu speed: 30...
>>3
Input index>>3
Not equal in second task

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>4
-111 -123 -1
-2

Maximal value: 0
1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>_
```

И чуть не забыл про транспонирование матрицы:

```
DOSBox 0.74-3, Cpu speed: 30...
6. exit
>>1
  1   2   3   4
  5   6   7   8
  9  10  11  12

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>5
  1   5   9
  2   6  10
  3   7  11
  4   8  12

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>_
```

Тестовый пример с большой матрицей: 7x8

```
DOSBox 0.74-3, Cpu speed: 30...
3
-4

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>1
  1   2   3   4   5   6   7   8
  9  10  11  12  13  14  15  16
 17  18  19  20  21  22  23  24
 -1  -2  -3  -4  -5  -6  -7  -8
 -9 -999 -100 -200 300 400 500 33
-33 -23  22 332  23   0   0  22
  3   4   5   6  -1  -2   3  -4

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
>>
```

DOSBox 0.74-3, Cpu speed: 30...

2	10	18	-2	4	1	18
3	11	19	3	12	10	3
4	12	2	11	2	19	2
5	1	10	19	11	11	11
6	9	18	3	2	11	-2
7	15	23	-7	500	0	7
8	16	24	-8	33	6	15

```
1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
```

```
>>3
```

```
Input index>>2
```

```
Not equal in second task
```

```
1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
```

```
>>_
```

DOSBox 0.74-3, Cpu speed: 30...

```
Not equal in second task
```

```
1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
```

```
>>4
```

```
2 3 4 5 6 7
```

```
11 12 1 9
```

```
2 10
```

```
Maximal value: 12
```

```
1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit
```

```
>>
```

DOSBox 0.74-3, Cpu speed: 30...

3 4 5 6 -1 -2 3 -4

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit

>>5

1	9	17	-1	-9	5	9
2	10	18	-2	4	1	18
3	11	19	3	12	10	3
4	12	2	11	2	19	2
5	1	10	19	11	11	11
6	9	18	3	2	11	-2
7	15	23	-7	500	0	7
8	16	24	-8	33	6	15

1. print matrix
2. move zeros
3. compare row and column
4. find max
5. trans
6. exit

>>