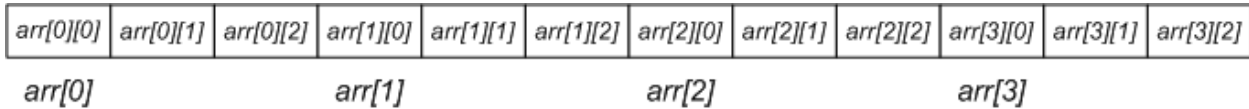



```
//
```

```
a[3][0] a[3][1] a[3][2]
```

Елементи багатовимірних масивів розташовуються в пам'яті підряд по рядках, тобто швидше за все змінюється останній індекс, а повільніше – перший.

Тобто масив *arr* з останнього прикладу в пам'яті буде розміщений таким чином:



Такий порядок розміщення дає можливість звертатися до будь-якого елементу багатовимірного масиву, використовуючи адресу його початкового елемента і тільки один індексний вираз.

Наприклад: звернення до елемента *arr*[1][2] можна здійснити за допомогою вказівника *ptr*.

```
int arr[4][3];
int *ptr =&arr[0][0]; // вказівник ptr вказує на перший елемент масиву arr
int b1 = arr[1][2]; // змінної b1 присвоюється значення arr[1][2]
int b2 = ptr[1*3+2]; // змінної b2 присвоюється значення arr[1][2]
int b3 = *(ptr + 1*3 + 2); // змінної b3 присвоюється значення arr[1][2]
```

Зазвичай, розмірності масивів оголошуються на початку файлу у вигляді констант за допомогою директиви препроцесора *#define*. Імена констант, зазвичай, пишуться великими літерами. Використання констант в якості розмірностей масивів дозволяють полегшувати модифікацію програм.

Наприклад:

```
#define N 5
#define M 4

int main()
{
    int a[N][M];
```

```
} ...
```

У даному прикладі оголошується двовимірний масив з розмірність 5 на 4.

Багатовимірні масиви ініціалізуються так само, як і одномірні.

Наприклад:

```
int w[3][3] = { { 2, 3, 4 },  
               { 3, 4, 8 },  
               { 1, 0, 9 } };
```

В останньому прикладі оголошений масив $w[3][3]$. Списки, виділені в фігурні дужки, відповідають рядкам масиву.

Наприклад:

```
int w[3][5] = {{1, 2, 3}, {4, 5, 6, 7, 8}, {9, 10, 11}};
```

ініціалізація виконається таким чином:

1	2	3		
4	5	6	7	8
9	10	11		

У разі відсутності дужок ініціалізація може бути виконана неправильно.

Наприклад:

```
int w[3][5] = {1, 2, 3, 4, 5, 6,7, 8, 9, 10, 11};
```

ініціалізація виконається таким чином:

1	2	3	4	5
6	7	8	9	10
11				

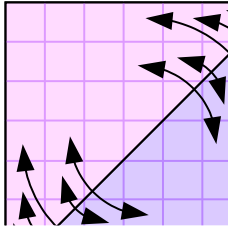
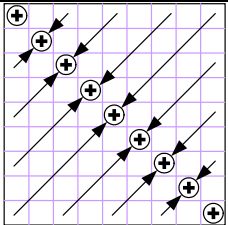
Робоче завдання

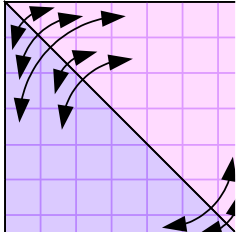
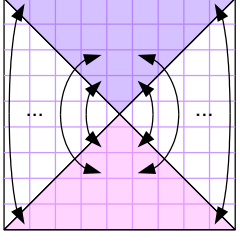
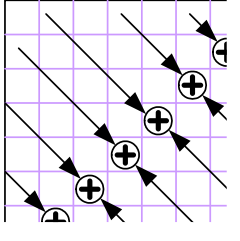
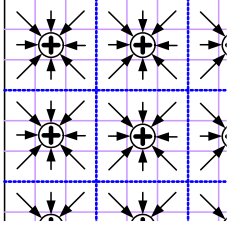
Скласти програму, в якій:

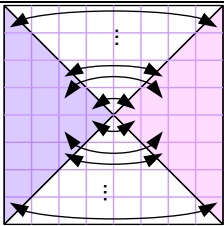
1. Згенерувати матрицю цілих чисел розмірністю згідно варіанту.
2. Елементи матриці задати випадковим чином в діапазоні 0 ... 100*.
3. Виконати форматний друк цієї матриці на екран.
4. Виконати обробку матриці згідно варіанту. При написанні програми використати універсальні алгоритми, який буде легко модифікувати під масиви іншої розмірності.
5. Виконати форматний друк перетвореної матриці на екран (за винятком варіантів № 9 та 19).

* Для цілей налагодження програми можна користуватися наперед проініціалізовану матрицею.

Номер	Розмір матриці	Завдання
1	9 x 9	Видалити з матриці рядок і стовпець, в яких розташований мінімальний по модулю елемент матриці. ПРИМІТКА. Якщо в матриці є кілька елементів з мінімальним по модулю значенням, то використовувати останній з них.
2	8 x 8	Знайти максимальний по модулю елемент матриці. Переставити рядки і стовпці матриці таким чином, щоб максимальний по модулю елемент був розташований на перетині 5-го рядка і 3-го стовпця. ПРИМІТКА. Якщо в матриці є кілька елементів з максимальним по модулю значенням, то використовувати останній з них.

Номер	Розмір матриці	Завдання
3	7 x 7	 <p>Відобразити матрицю симетрично відносно побічної діагоналі.</p>
4	9 x 9	Піднести матрицю до квадрата (помножити саму на себе).
5	8 x 8	<p>Виключити з матриці рядок і стовпець, на перетині яких розташований максимальний елемент головної діагоналі.</p> <p>ПРИМІТКА. Якщо в матриці є кілька елементів з максимальним значенням, то використовувати останній з них.</p>
6	8 x 8	 <p>Поміняти місцями елементи головної і побічної діагоналей.</p>
7	9 x 9	 <p>Розмістити на головній діагоналі суми елементів, які лежать на діагоналях, перпендикулярних до головної (включати в суму значення елемента на головній діагоналі).</p>
8	8 x 8	<p>Переставити рядок з максимальним по модулю елементом на головній діагоналі з сусіднім нижнім рядком.</p> <p>ПРИМІТКА. Якщо максимальний по модулю елемент головної діагоналі знаходиться в самій нижній рядку, то переставляти потрібно з сусідньої верхнім рядком.</p>
9	9 x 9	Елемент матриці називається локальним мінімумом, якщо він строго менше всіх наявних у нього сусідів. Підрахувати кількість локальних мінімумів заданої матриці. Роздрукувати їх значення і положення (номера рядка і стовпця, де розташовується цей елемент).

Номер	Розмір матриці	Завдання
10	8 x 8	 <p>Відобразити матрицю симетрично відносно головної діагоналі.</p>
11	7 x 7	<p>Знайти мінімальний по модулю елемент матриці. Переставити рядки і стовпці матриці таким чином, щоб мінімальний по модулю елемент був розташований на перетині 2-го рядка і 6-го стовпця.</p> <p>ПРИМІТКА. Якщо в матриці є кілька елементів з мінімальним по модулю значенням, то використовувати останній з них.</p>
12	7 x 7	 <p>Відобразити симетрично відносно горизонтальної осі сектори матриці, які лежать вище та нижче головної та побічної діагоналей.</p>
13	8 x 8	 <p>Розмістити на побічній діагоналі суми елементів, які лежать на діагоналях, перпендикулярних до побічної.</p>
14	7 x 7	Піднести матрицю до куба.
15	9 x 9	 <p>Розбити матрицю на квадрати розміром 3x3. У центрі кожного квадрата помістити суму інших елементів квадрата.</p>
16	8 x 8	<p>Виключити з матриці рядок і стовпець, на перетині яких розташований мінімальний елемент головної діагоналі.</p> <p>ПРИМІТКА. Якщо в матриці є кілька елементів з мінімальним значенням, то використовувати останній з них.</p>

Номер	Розмір матриці	Завдання
17	9 x 9	 <p>На головній діагоналі розмістити суми елементів, які лежать на тому ж рядку і стовпці.</p>
18	7 x 7	Виконати транспонування матриці.
19	9 x 9	Елемент матриці називається локальним максимумом, якщо він строго більше за всіх наявних у нього сусідів. Підрахувати кількість локальних максимумів заданої матриці. Роздрукувати їх значення і положення (номера рядка і стовпця, де розташовується цей елемент).
20	9 x 9	<p>Переставити стовпець з мінімальним елементом на головній діагоналі з сусіднім правим стовпцем.</p> <p>ПРИМІТКА. Якщо мінімальний елемент головної діагоналі знаходиться в крайньому правому стовпці, то переставляти потрібно з сусіднім лівим стовпцем.</p>
21	8 x 8	 <p>Показати симетрично відносно горизонтальної осі сектори матриці, які лежать правіше і лівіше головної та побічної діагоналей.</p>

Контрольні питання

1. Чи існує обмеження на кількість вимірів масиву?
2. Як розміщуються в пам'яті багатомірні масиви?
3. Які способи звернення до елементів багатомірного масиву ви знаєте?
4. Чи припустимо опустити одну з розмірностей масиву? Якщо так, то яку в яких ситуаціях?
5. Як можна проініціалізувати багатомірні масиви?