

Практичне завдання № 1-1

ДОСЛІДЖЕННЯ РОБОТИ З БАГАТОВИМІРНИМИ МАСИВАМИ

Мета завдання: ознайомитися з базовими елементами мови Java та навчитися коректно оброблювати набори даних у вигляді масивів.

Завдання: Напишіть програму на мові Java, яка виводить на екран прізвище та ініціали розробника, а потім виконує обробку матриці $A[N][N]$ відповідно варіанту завдання (таблиця). Значення елементів матриці $A[N][N]$ задаються за допомогою генератора випадкових чисел. Результат виводиться на консоль.

Варіант обирається відповідно списку групи.

Варіанти

Номер варіанту	Вимірність (N)	Тип елементу	Завдання
1.	5	short	Упорядкуйте рядки матриці за зростанням значень елементів k -го стовпця
2.	7	int	Виконайте циклічний зсув матриці на k позицій вліво
3.	6	double	Знайдіть суму елементів матриці, які розташовані між першим та другим від'ємними елементами
4.	7	int	Поміняйте місцями елементи першого стовпця з останнім, другого – з передостаннім і т.д.
5.	5	byte	Виконайте транспонування матриці
6.	7	long	Упорядкуйте стовпці матриці за зростанням значень елементів m -го рядка
7.	6	double	Виконайте циклічний зсув матриці на m позицій до низу
8.	6	float	Виконайте поворот матриці на 90° проти годинникової стрілки
9.	5	long	Змініть значення елементів матриці відніманням від кожного елементу рядка середнє арифметичне цього рядка
10.	7	short	Знайдіть максимальний елемент матриці та видаліть з неї стовпець з цим елементом.
11.	5	float	Виконайте циклічний зсув матриці на n позицій вправо
12.	5	int	Переставте елементи рядків матриці таким чином, щоб елементи з нульовим значенням розташовувалися наприкінці рядка
13.	7	byte	Знайдіть мінімальний елемент матриці та розташуйте його за вказаною позицією шляхом послідовного зсуву (обміну) рядків і стовпців.
14.	6	short	Упорядкуйте стовпці матриці за спаданням значень елементів k -го рядка
15.	5	double	Виконайте поворот матриці на 90° за годинниковою стрілкою
16.	6	byte	Виконати циклічний зсув матриці на n позицій вгору
17.	6	int	Знайдіть максимальний елемент матриці та видаліть з неї рядок з цим елементом
18.	6	short	Упорядкуйте рядки матриці за спаданням значень елементів k -го стовпця

Номер варіанту	Вимірність (N)	Тип елементу	Завдання
19.	7	double	Виконайте округлення всіх елементів матриці до цілого значення за правилом в математиці
20.	5	byte	Поміняйте місцями елементи першого рядка з останнім, другого - з передостаннім і т.д.
21.	6	int	Виконайте поворот матриці на 180° проти годинникової стрілки
22.	7	long	Знайдіть мінімальний елемент матриці та видаліть з неї рядок з цим елементом
23.	5	double	Змініть значення елементів матриці відніманням від кожного елементу стовпця середнє арифметичне цього стовпця
24.	6	byte	Переставте елементи рядків матриці таким чином, щоб елементи з нульовим значенням розташовувалися на початку рядка
25.	7	int	Знайдіть максимальний елемент матриці та розташуйте його за вказаною позицією шляхом послідовного зсуву (обміну) рядків і стовпців.
26.	5	short	Обчисліть середнє арифметичне елементів матриці та змініть елементи головної діагоналі на це значення.
27.	6	double	Знайдіть максимальний елемент матриці та поміняйте його місцями з нижнім правим елементом
28.	7	byte	Знайдіть мінімальний елемент матриці та видаліть з неї стовбець з цим елементом
29.	5	int	Обчисліть середнє арифметичне елементів матриці та змініть елементи побічної діагоналі на це значення.
30.	6	float	Знайдіть мінімальний елемент матриці та поміняйте його місцями з верхнім лівим елементом

Методичні вказівки

Для демонстрації роботи програми з матрицею різної вимірності, забезпечити визначення значення N набором з клавіатури.

Для наочності виконання програми необхідно вивести на екран матрицю $A[N][N]$ до її обробки та після (для кожного рядка матриці – окремий рядок, стовпці матриці повинні бути вирівняними). Для цього можна використати вираз **System.out.printf(...)**, тобто форматове виведення даних.

Для ініціалізації елементів матриці необхідно використовувати метод *random()* класу **Math**. Наприклад, для отримання випадкового цілого числа з діапазону $[0; 100]$ використовуйте наступний вираз: `(int)(Math.random()*100)`. Також для цього можна використовувати екземпляр класу **Random** та його метод *nextInt(100)*.