

Two Dimensional arrays

1. Прочетете число **N** от конзолата – брой играчи. След това прочетете **N** числа – най-високи точки на всеки от играчите в дадена игра(винаги са подредени така, че най-високите да са в началото). Покажете точките на първите трима.

Вход	Изход
5 3100 2700 1500 400 100	3100 2700 1500

2. Прочетете число **N** от конзолата – брой на играчите, които ще бъдат въведени. След това прочетете **Nx3** числа – за всеки играч прочетете по 3 числа – трите му най-високи точки. Покажете най-високите точки за всеки играч.

Вход	Изход
3 2100 3700 1400 3500 3010 2900 1500 1500 1501	3700 3500 1501

3. Прочетете две числа **n** и **m** от конзолата, а след това прочетете **n** реда по **m** числа. Покажете сумата им. Използвайте двумерен масив, за да решите задачата.

Вход	Изход
3 3 1 2 3 4 5 6 7 8 9	45

4. Прочетете три числа **N**, **M** и **K** от конзолата, създайте матрица **NxM** и я напълнете с числото **K**. Покажете резултата на конзолата.

Вход	Изход
3 4 5	5 5 5 5 5 5 5 5 5 5 5 5

Two Dimensional arrays

5. Прочете матрица от конзолата, изведете сумата на двата и диагонала.

Вход	Изход
3 3 1 0 1 0 1 0 1 2 3	8

- Решете задачата като броите всеки елемент само веднъж.

6. Напишете функции, които :

- намират сумата на дадена **колона** в двумерен масив(матрица)
- Намират сумата на даден **ред** в матрица
- Намират всички двойки индекси на редове и съответно колони, чиито суми са еднакви
 - Какви параметри трябва да приемат функциите?
 - Как ще върнем повече от 1 стойност от функция? Повече от 2? Повече от 5?

7. Една матрица е **row-magical**, ако всеки ред има еднаква сума. Съответно **col-magical**, ако всяка колона има една и съща сума. Напишете функции, които проверяват дали една матрица е **row-magical** или **col-magical**.

8. Напишете програма, която приема число **N** – дължина на редица от число, а след това **N** числа. Да се покаже редицата в спираловидна форма. Може да приемете, че **N** е във вида **$N = k^2$** , където **k** е естествено, положително число(това значи че матрицата, която трябва да използвате е **$N \times N$** , казано по сложен начин).

Вход	Изход
4 1 2 3 4	1 2 4 3
9 1 2 3 4 5 6 7 8 9	1 2 3 8 9 4 7 6 5

Задачи за едномерни масиви:

- [10 - 15 от предишното упражнение](#)
- [Задачите от допълнителното упражнение](#)