## Задача 1 – Quadronacci правоъгълник

### Всички знаем редицата на Fibonacci, където всяко число е сумата от предишните две. Е, Quadronacci редица е почти същата, но използва предишните четири числа, за да се изчислява. Накратко можем да дефинираме, че n-тото число от редицата е:

Q n = Q n-1 + Q n-2 + Q n-3 + Q n-4

където Q n е текущото Quadronacci число (n е индекса на n-тото Quadronacci число).

Quadronacci редицата може да започне с които и да е четири цели числа, положителни или отрицателни, и да продължи по дефинираната формула.

От друга страна, Quadronacci правоъгълник е нещо, което може би очаквате – правоъгълник (матрица) от числа, които се съдържат в Quadronacci редицата. Височината на правоъгълника е броя редове в матрицата, а широчината – броя колони.

Нека **R** е броя редове, а **C** – броя колони. Тогава първият ред на правоъгълника ще съдържа първите **C** числа от редицата, втория ред ще съдържа следващите **C** числа от редицата и т.н.

Задачата ви е да напишете програма, която принтира на конзолата Quadronacci правоъгълника при дадени първите четири числа от редицата, броя редове и броя колони в матрицата.

### Входни данни

Входните данни ще се четат от конзолата.

**Първите четири реда** ще съдържат стойностите на **първите четири цели числа** от Quadronacci редицата – всяко число на различен ред.

Петият ред ще съдържа числото **R** – броят редове в правоъгълника.

Шестият ред ще съдържа числото **C** – броят колони в правоъгълника.

Входните данни ще бъдат винаги валидни и в описания формат. Не е нужно да бъдат проверявани изрично.

### Изходни данни

Изходните данни ще се отпечатват на конзолата.

Изхода трябва да съдържа точно R на брой линии, съдържащи точно C на брой числа – редицата на Quadronacci, като между всеки две числа трябва да има точно един интервал.

### Ограничения

* 1 ≤ **R** ≤ 20.
* 4 ≤ **C** ≤ 20.
* Всяко число от редицата може да се събере в 64 битов целочислен тип.
* Позволено време: **0.1 секунда**. Позволена памет: **16 MB**.

### Примери

|  |  |
| --- | --- |
| **Вход** | **Изход** |
| 1  2  3  4  2  8 | 1 2 3 4 10 19 36 69  134 258 497 958 1847 3560 6862 13227 |
| 5  -5  1  2  3  4 | 5 -5 1 2  3 1 7 13  24 45 89 171 |