

## 3. Stamp

### Zadanie

Dana jest tablica kwadratowa  $T$  o boku  $n$  zawierająca wartości całkowite oraz tablica prostokątna  $P$  o wymiarach  $k \times l$  przechowująca wartości 0 lub 1. Tablice  $P$  możemy “nałożyć” na tablicę  $T$  tak, by przykryła ona pewien jej fragment (ale musi mieścić się całkowicie w obrębie  $T$ ). Szukamy takiego położenia tablicy  $P$ , że suma elementów  $T_{ij}$  przykrytych przez elementy  $P$  o wartości 1 była największa.

Napisz program, który:

1. Wczyta rozmiar tablicy  $T$ ,  $n$ , rozmiary tablicy  $P$ ,  $k$  i  $l$  a następnie tablice  $T$  i  $P$ ,
2. Wyznaczy optymalne położenie  $P$ ,
3. Wypisze maksymalną sumę elementów  $T$  przykrytych przez elementy  $P$  równe 1.

### Wejście:

W pierwszym wierszu standardowego wejścia znajdują się trzy liczby całkowite  $1 \leq n \leq 100$  i  $1 \leq k, l < n$ . Kolejne  $n$  wierszy zawiera po  $n$  liczb całkowitych (wiersze  $T$ ). Następne  $k$  wierszy (po  $l$  elementów 0 lub 1) stanowią wiersze tablicy  $P$ .

### Wyjście:

Program powinien wypisać jedną liczbę całkowitą - największą sumę “przykrytych” wartości tablicy  $T$ .

### Przykład:

Dla danych wejściowych:

```
5 2 2
5 1 0 1 0
2 3 4 7 1
2 7 4 0 1
1 8 5 1 9
0 1 6 3 0
0 1
1 0
```

poprawną odpowiedzią jest:

12