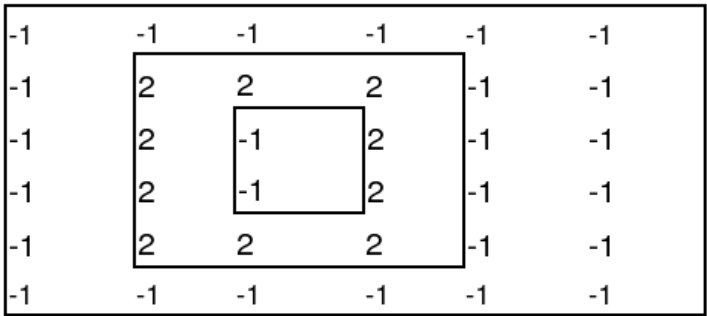


## 11263 Nested Rectangles

Sultan has a rectangle of  $R$  rows and  $C$  columns. Each cell of this rectangle contains an integer. Sultan chooses  $n$  subrectangles. The  $i$ -th subrectangle has  $R_i$  rows and  $C_i$  columns and it is nested inside  $(i - 1)$ -th subrectangle. The first subrectangle is nested inside the initial rectangle. Sultan then multiplies all the integers outside the first subrectangle with  $M_0$ . Then he multiplies all the integers inside  $i$ -th rectangle but outside  $(i + 1)$ -th rectangle with  $M_i$ . Then he multiplies all the integers inside  $n$ -th subrectangle with  $M_n$ . So he get a new rectangle of integers. The sum of all the integers of this new rectangle is  $S$ . Help Sultan to choose all this subrectangles in such a way so that  $S$  is maximized.



In the above figure, the outer most portion (that is not contained in any of the sub rectangle) is multiplied by  $M_0$ , the portion inside the first rectangle, but outside the second one by  $M_1$ , portion inside 2nd and outside 3rd by  $M_2$ , and so forth. The portion inside the  $n$ -th sub rectangle is multiplied by  $M_n$ .

### Input

First line of the input contains  $T (\leq 20)$  the number of test cases. First line of the each test case contains 3 integers  $R (1 \leq R \leq 500)$ ,  $C (1 \leq C \leq 500)$  and  $n (1 \leq n \leq 5)$ . Second line contains  $n$  integers  $R_1, R_2, \dots, R_n (R > R_1 > R_2 > \dots > R_n)$ . Third line contains  $n$  integers  $C_1, C_2, \dots, C_n (C > C_1 > C_2 > \dots > C_n)$ . The values  $R_i, C_i$  describes the dimensions of the  $i$ -th sub rectangle. Fourth line contains  $n + 1$  integers  $M_0, M_1, \dots, M_n (-10 \leq M_i \leq 10)$ , the values of each multiplier. Lines 5 to line  $4 + R$  each contain  $C$  integers. The  $j$ -th integer in the  $(i + 4)$ -th line is the number in the  $i$ -th row and  $j$ -th column of the initial rectangle. All the integers in the initial rectangle is between -100 to +100 inclusive.

### Output

For each test case output contains one integer denoting the maximum value of  $S$ .

### Sample Input

```

1
6 6 2
4 2
3 1
0 1 -1
-1 -1 -1 -1 -1 -1

```

```
-1  2  2  2 -1 -1
-1  2 -1  2 -1 -1
-1  2 -1  2 -1 -1
-1  2  2  2 -1 -1
-1 -1 -1 -1 -1 -1
```

**Sample Output**

22