

## Tortoise Chess

### Problem Description

On Xiao Ming's birthday, his father gave him a set of tortoise chess as a gift. The board has  $N$  squares in one line, each with a score (a non-negative integer). The first square is the only starting point and the  $N^{\text{th}}$  grid is the endpoint. The game requires the player to control a tortoise piece to move from the starting point to the endpoint.

In tortoise chess, there are  $M$  crawling cards, which are divided into 4 different types (not necessarily all 4 types of cards are included in  $M$  cards, see the sample). Each type of card is marked with one of the four numbers 1, 2, 3 and 4, indicating the corresponding number of squares that the tortoise piece will crawl forward after using this kind of card. In the game, the player needs to select an unused crawling card from all the crawling cards and control the tortoise piece to advance the corresponding number of squares, each card can only be used once.

In the game, the tortoise piece automatically obtains the points of the starting square and gets the corresponding points of the square when it reaches a square in the subsequent crawl. The final score of the player is the sum of the points of all the squares that the tortoise piece visited during its journey from beginning to end.

Obviously, using different orders of cards will result in different scores. Xiao Ming wants to find an order of cards that will get the most score in the game.

Now, given the score of each square on the board and all the crawling cards, can you tell Xiao Ming the maximum number of points he can score?

### Input

Every two numbers are separated by a space.

The first line contains two positive integers  $N$  and  $M$ , representing the number of chessboard squares and the number of crawl cards.

The second line contains  $N$  non-negative integers,  $a_1, a_2, \dots, a_N$ , where  $a_i$  represents the score on the  $i^{\text{th}}$  square of the chessboard.

The third line contains  $M$  integers,  $b_1, b_2, \dots, b_M$ , representing the numbers on the  $M$  crawling cards.

The input data is guaranteed to use up exactly  $M$  crawl cards when reaching the endpoint.

### Output

There is an integer indicating the maximum number of points Xiao Ming can get.

**Sample Input**

9 5  
6 10 14 2 8 8 18 5 17  
1 3 1 2 1

**Sample Output**

73

**Hint**

1s for each test point

Xiao Ming uses the crawling card order 1, 1, 3, 1, 2 and obtains a score of  $6+10+14+8+18+17=73$ . Note that since the starting point is 1, the 6 points from the 1st square are automatically obtained.

For 30% of the data,  $1 \leq N \leq 30$ ,  $1 \leq M \leq 12$ .

For 50% of the data,  $1 \leq N \leq 120$ ,  $1 \leq M \leq 50$ , and there are 4 kinds of crawling cards. The number of cards for each kind will not exceed 20.

For 100% of the data,  $1 \leq N \leq 350$ ,  $1 \leq M \leq 120$ , and there are 4 kinds of crawling cards. The number of cards for each kind will not exceed 40;  $0 \leq a_i \leq 100$ ,  $1 \leq i \leq N$ ,  $1 \leq b_i \leq 4$ ,  $1 \leq i \leq M$ .