

# Rolling For Days

Input file:            **standard input**  
Output file:          **standard output**  
Time limit:           2 seconds  
Memory limit:        1024 megabytes

In the game TFT, players need to buy units in the store to build up a powerful lineup against other players. Only one-star units will be refreshed in the store, three identical one-star units can be synthesized into a corresponding two-star unit, and three identical two-star units can be synthesized into a corresponding three-star unit. Higher star units will obviously be much more powerful than lower star units. The units refreshed in the store are random, and refreshing requires coins. So it's possible that you can't get the units you want although having a lot of coins.



An annoying example from the game

As a disciplined player, you're curious about just how many refreshes are needed to complete a particular deck. To that end, you're going to start by solving the following simplified problem:

There are a total of  $n$  cards in the store's card pool, which are categorized into  $m$  kinds, where the  $i$ -th kind of card has a total of  $a_i$  cards. You have decided on your target deck, and your target deck requires  $b_i$  cards of the  $i$ -th kind. Each time you refresh, one card will be randomly drawn from the remaining cards in the card pool with equal probability. If the drawn card is not needed (i.e., the number of cards in this kind already purchased is equal to the number that your target deck requires), you'll put the card right back into the card pool; otherwise, you'll purchase the card and remove it from the card pool. How many refreshes do you expect to need to complete your target deck? Since the answer may be large and floating point errors cannot be ignored, please output the answer modulo 998244353.

Formally, let  $M = 998244353$ . It can be shown that the answer can be expressed as an irreducible fraction  $\frac{p}{q}$ , where  $p$  and  $q$  are integers and  $q \not\equiv 0 \pmod{M}$ . Output the integer equal to  $p \cdot q^{-1} \pmod{M}$ . In other words, output such an integer  $x$  that  $0 \leq x < M$  and  $x \cdot q \equiv p \pmod{M}$ .

## Input

The first line contains two integers  $n, m$  ( $1 \leq n \leq 1000; 1 \leq m \leq 12$ ), denoting the total number of cards and the number of kinds.

The second line contains  $m$  integers  $a_1, a_2, \dots, a_m$  ( $1 \leq a_i \leq n, \sum_{i=1}^m a_i = n$ ), denoting the numbers of cards of each type.

The third line contains  $m$  integers  $b_1, b_2, \dots, b_m$  ( $0 \leq b_i \leq a_i$ ), denoting the numbers of cards required for your target deck.

## Output

Output a single integer, denoting the expect number of refreshes modulo 998244353.

## Examples

standard input	standard output
2 2 1 1 1 1	2
4 2 2 2 2 1	582309210
1000 12 101 43 34 281 23 24 12 25 66 222 145 24 37 43 27 257 5 11 12 19 62 41 87 13	265294941