

## Problem A. 2016

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

2016 .

$d(n)$   $n$  . ,  $d(12) = 6$ ,  $12 : 6 : 1, 2, 3, 4, 6, 12$ .

$x$  ,  $y$  ,  $y < x$   $d(y) > d(x)$  . , 2016 , , 2016, 1680 , 2016.

$K$ .  $K$ - .  $10^{18}$ ,  $-1$ .

### Input

$K$  ( $1 \leq K \leq 10^9$ ).

### Output

— .

### Examples

standard input	standard output
10	14
10000000000	-1

### Note

1, 2, 3, 4, 5, 6, 8, 10, 12, 14, ...

## Problem B. Airports

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

Snuke —  $N$  .  
 $i$ - —  $(x_i, y_i)$ . Snuke  $D$   $p$   $q$  ,  $p$   $q$   $D$ .  $D$  .  
,  $(x_1, y_1)$   $(x_2, y_2)$   $|x_1 - x_2| + |y_1 - y_2|$ .

### Input

$N$  ( $2 \leq N \leq 10^5$ ).  $N$  ,  $i$ -  $x_i$   $y_i$  —  $i$ - ( $0 \leq x_i, y_i \leq 10^9$ ). .

### Output

.

### Example

standard input	standard output
6 1 7 8 5 6 3 10 3 5 2 6 10	9

# Problem C. Jump

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

Snuke has  $N$  integers  $a_1, a_2, \dots, a_N$ . He will choose an integer  $x$  and then, for each  $i$  ( $1 \leq i \leq N$ ), he will perform the following operation  $s_i$  times:  $a_i \leftarrow a_i - x$ . He wants to minimize the maximum value of  $a_i$  after the operations. Find the minimum possible value of the maximum value of  $a_i$ .

## Input

The first line contains  $N$  ( $1 \leq N \leq 200$ ). The second line contains  $a_1, a_2, \dots, a_N$  ( $0 \leq a_1 < \dots < a_N \leq 10^4$ ). The third line contains  $Q$  ( $0 \leq Q \leq 10^5$ ). The next  $Q$  lines contain  $s_i$  and  $t_i$  ( $0 \leq s_i, t_i \leq 10^4$ ).

## Output

## Example

standard input	standard output
4	-1
1	-1
2	2
4	2
7	-1
10	-1
2 3	0
5 6	3
6 0	1
3 7	0
10 3	
7 6	
5 5	
2 10	
4 10	
10 10	

# Problem D. Merge

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

Snuke has  $R$  red stones,  $P$  purple stones, and  $Q$  blue stones. He will perform the following operation  $R$  times:

- Choose a red stone.
- Choose a purple stone or a blue stone (if both are available).

Let  $P_i$  and  $Q_i$  be the number of purple and blue stones remaining after the  $i$ -th operation, respectively. Snuke wants to maximize the value of  $\sum_{i=1}^R (P_i + Q_i)$  modulo  $10^9 + 7$ .

## Input

The first line contains an integer  $N$  ( $1 \leq N \leq 2000$ ). The second line contains  $N$  integers  $P_1, P_2, \dots, P_N$  ( $1 \leq P_i \leq N$ ,  $P_i \neq P_j$  for  $i \neq j$ ). The third line contains  $N$  integers  $Q_1, Q_2, \dots, Q_N$  ( $1 \leq Q_i \leq N$ ,  $Q_i \neq Q_j$  for  $i \neq j$ ).

## Output

Print the maximum value of  $\sum_{i=1}^R (P_i + Q_i)$  modulo  $10^9 + 7$ .

## Examples

standard input	standard output
4 3 1 2 4 3 1 2 4	14
10 5 7 3 1 6 4 2 10 9 8 2 8 9 1 5 6 10 4 3 7	127224

# Problem E. Mirror Rice Cake

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

( ) — , .  
Snuke  $N$  .  
 $i$ -  $a_i$  . , , , .  
Snuke , .

## Input

$N$  ( $1 \leq N \leq 1000$ ).  $N$   $a_i$  ( $1 \leq a_i \leq 10^9$ ).

## Output

— .

## Example

standard input	standard output
5 3 20 5 8 6	3

## Note

3, 5, 20 .

## Problem F. Number Cards

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

Snuke has  $N$  cards. The  $i$ -th card has the number  $a_i$  and the cost  $c_i$ .

Snuke wants to choose a subset of these cards such that

- $1 \leq a_i \leq M$  for all  $i$ .
- $M + 1 \leq a_i \leq 2M$  for all  $i$ , and  $1 \leq a_i \leq M$  for all  $i$ .
- $2M + 1 \leq a_i \leq 3M$  for all  $i$ , and  $1 \leq a_i \leq 2M$  for all  $i$ .
- $3M + 1 \leq a_i \leq 4M$  for all  $i$ , and  $1 \leq a_i \leq 3M$  for all  $i$ .
- ...

where  $M$  is a positive integer.

### Input

The first line contains  $N$  ( $1 \leq N \leq 20$ ). The next  $N$  lines contain  $a_i$  and  $c_i$  separated by a space, where  $1 \leq a_i \leq 10^9$ ,  $1 \leq c_i \leq 20$ .  
The  $i$ -th line contains  $a_i$  and  $c_i$ .

### Output

Print the maximum value of  $\sum c_i$  for the chosen subset of cards.

### Examples

standard input	standard output
4 27 2 2000 4 2015 4 2100 1	277
3 1 1 2 2 3 1	0

## Problem H. Random Walk

Input file: *standard input*  
Output file: *standard output*  
Time limit: 3.5 seconds  
Memory limit: 512 mebibytes

. Snuke :  
 $(0, 0) \rightarrow N \cdot (i, j), \quad \text{---} \quad (i - 1, j), (i, j - 1), (i, j + 1), (i + 1, j). \quad \frac{1}{4}.$   
 $E \text{ --- } E \times 4^N \quad M (, ). , (0, 0) .$

### Input

$N \quad M \quad (1 \leq N \leq 5000, 10^9 \leq M \leq 2 \times 10^9).$

### Output

.

### Examples

standard input	standard output
2 1000000007	44
2015 20000000000	1892319232

## Problem I. Robots

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

Snuke  $N$  .  $1 \leq N$ .

$i$   $(x_i, y_i)$   $d_i$  ( $d_i$  — 'U', 'D', 'L' 'R' —  $y, y, x, x$  .  
 . - ( Snuke), , , . ( , ).

Snuke  $1 \leq 0$ .  $T$ .

### Input

$N \leq T$  ( $1 \leq N \leq 10^5$ ,  $0 \leq T \leq 10^{18}$ ).  $i$ -  $N$   $x_i \ y_i \ d_i$  —  $i$ - ( $0 \leq x_i, y_i \leq 10^9$ ,  $d_i$  — 'U', 'D', 'L', 'R').  $0 \leq d_i$  .

### Output

$N$  .  $i$ -  $i$ -  $T$ .

### Example

standard input	standard output
5 10	1 10
1 0 U	3 6
3 1 U	9 2
1 2 R	-8 1
1 1 L	8 1
0 1 R	



# Problem J. Ropes

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

$N$  . 1  $N$ . Snuke ,  $N - 1$  , :

- ;
- ( ).
- $i$ -  $a_i$  .

, . , , .

## Input

$N$  ( $2 \leq N \leq 10^5$ ).  $i$ -  $N$   $a_i$  — ,  $i$ - ( $1 \leq a_i \leq 3$ ).

## Output

— .

## Example

standard input	standard output
9 1 3 2 1 3 1 2 1 2	1260

## Problem L. String Modification

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

Snuke has a string  $s$ . Snuke can perform the following operation any number of times (possibly zero):

Choose an index  $i$  and a character  $c$ .

Replace the character at index  $i$  in  $s$  with  $c$ . For example, if  $s = \text{"abca"}$  and  $i = 2$ , then after the operation  $s = \text{"adbca"}$ .

### Input

The input consists of two lines. The first line contains the string  $s$ , and the second line contains the string  $t$ . It is guaranteed that  $1 \leq |s| \leq |t| \leq 5000$ .

### Output

Print "Yes" if Snuke can transform  $s$  into  $t$ , and "No" otherwise.

### Examples

standard input	standard output
snuke snukent	Yes
snuke ssnuke	No

# Problem N. Soccer Match

Input file: *standard input*  
Output file: *standard output*  
Time limit: 2 seconds  
Memory limit: 512 mebibytes

$n, m, k, l, w, t$  — 0.  
 $n, m, k, l, w, t$  —  $N, K$ .  
( $n, m, k, l, w, t$ ).

## Input

$N$  ( $0 < N \leq 100$ )  $K$  ( $0 \leq K \leq 300$ ) —  
 $n, m, k, l, w, t$  —  $N, K$ .

## Output

$n, m, k, l, w, t$  —  $w, t, l$ .

## Example

standard input	standard output
6 10	3 1 2 2 4 0