

## Problem Set

A. 배수들의 합 .....	1
B. 전체 계산 횟수 .....	2
C. 피보나치는 지겨웁~ .....	3
D. 암호해독기 .....	4
E. 내접사각형 만들기 .....	5
F. 줄서기 .....	6-9
G. 케이크 자르기 .....	10
H. 문자열 비교하기 .....	11
I. 나랏말싸미 America와 different~ .....	12-13
J. 우주 탐사선 .....	14

# Problem A

## 배수들의 합

Time Limit: 1 Second

신원이는 백준에서 배수에 관한 문제를 풀다가 감명을 받아 새로운 문제를 만들어보았다. 자연수  $N$ 과  $M$ 개의 자연수  $K_i$ 가 주어진다.  $K_i$ 의 배수들 중  $N$  이하인 숫자의 합을 구하여라.

### Input

첫 번째 줄에  $N$ 과  $M$ 가 주어진다. ( $2 \leq N \leq 1000$ ,  $1 \leq M < N$ )  
그다음 줄에  $M$  개의 배수  $K_i$ 가 주어진다. ( $2 \leq K_i \leq 1000$ )  
동일한 배수는 주어지지 않으며, 오름차순으로 정렬되어 주어진다.

### Output

배수들의 합을 출력한다.  
단, 겹치는 배수는 한 번만 더한다.

#### Sample Input 1

10 2	Output for the Sample Input 1
2 3	42

#### Output for the Sample Input 1

#### Sample Input 2

1000 3	Output for the Sample Input 2
3 5 7	272066

#### Output for the Sample Input 2

## Problem B

### 전체 계산 횟수

Time Limit: 1 Second

호근이는 한국에서 미국으로 유학을 하러 갔다. 그는 생활비로 쓰기 위해 가지고 온 원화를 전부 달러로 바꾸기 위해 환전소를 찾아갔다. 그런데 환전소에 1달러짜리 지폐를 제외하고는 다른 지폐가 없어서 전부 1달러 지폐로 바꾸어 숙소로 가져왔다. 환전한 금액을 알고 있지만 불안한 마음에 숙소에서 다시 한 번 얼마인지 세어 보려고 한다. 이때 1달러 지폐를 한 장씩 세어서  $M$  개 짜리 묶음으로 만들고 이 묶음을 다시 한 번씩 세면서  $M$  개 짜리 묶음으로 만드는 방법을 반복하여 환전한 금액  $N$  원을 전부 세려고 한다. 이때 호근이가 묶음을 포함해 지폐를 센 전체 횟수를 구하여라.

예를 들어  $N$ 이 13이고  $M$ 이 10일 때, 13달러를 세기 위해서는 1달러씩 총 13번을 세고, 그 결과 10달러가 묶음이 1개 생기고, 3달러가 있다는 것을 알게 된다. 그다음 10달러 묶음을 1번 세야 하므로 총 14번을 세야 한다. 만들어진 묶음도 다시 세야 한다.

#### Input

첫 번째 줄에 환전한 금액  $N$  하고 묶음  $M$ 이 주어진다. ( $2 \leq N \leq 100,000$ ,  $2 \leq M \leq N$ )

#### Output

전체 센 횟수를 출력한다.

#### Sample Input 1

13 10

#### Output for the Sample Input 1

14

#### Sample Input 2

100 8

#### Output for the Sample Input 2

113

## Problem C

### 피보나치는 지겨웁~

Time Limit: 1 Second

혁신이는 알고리즘 문제를 만들라는 독촉을 받아 스트레스다. 하지만 피보나치 문제는 너무 많이 봐서 지겹기 그지없다. 그러나 문제를 만들 시간이 없는 혁신이는 피보나치 문제를 응용해서 문제를 만들려 한다.

```
int fibonacci(int n) { // 호출
    if (n < 2) {
        return n;
    }
    return fibonacci(n-2) + fibonacci(n-1);
}
```

위와 같이 코딩하였을 때 fibonacci(n)를 입력했을 때에 fibonacci 함수가 호출되는 횟수를 계산해보자.

#### Input

fibonacci 함수에 인자로 입력할  $N$ 이 주어진다. ( $0 \leq N \leq 50$ )

#### Output

fibonacci 함수가 호출된 횟수를 출력한다.

출력값이 매우 커질 수 있으므로 정답을 1,000,000,007로 나눈 나머지를 출력한다.

#### Sample Input 1

2

#### Output for the Sample Input 1

3

#### Sample Input 2

3

#### Output for the Sample Input 2

5

## Problem D

### 암호해독기

Time Limit: 1 Second

암호문이 하나 도착했다. 이것을 해석해서 무슨 의미인지를 유추해보고 주어진 해석문과 일치하는지 확인해보려 한다. 암호문은 0 - 52까지의 숫자가 띄어쓰기로 구분되어 주어진다. 해독된 암호문에서 0은 띄어쓰기에, 1 ~ 26까지의 숫자는 A ~ Z에, 27 ~ 52까지의 숫자는 a ~ z에 대응된다. 띄어쓰기를 포함한 모든 철자가 무작위로 뒤섞여 있는 암호문이다. 해독된 문장이 같이 주어질 때 주어진 암호문을 해독한 문장이 일치하는지를 확인하여라.

#### Input

첫 번째 줄에는 주어질 숫자열의 길이  $N$ 이 주어진다. ( $1 \leq N \leq 100,000$ )

두 번째 줄에는 암호문에 해당하는 숫자  $N$  개가 띄어쓰기와 함께 주어진다.

세 번째 줄에는 해독된 암호문이 주어진다.

단, 해독된 암호문의 길이는  $N$ 과 같으며, 암호문의 시작과 끝으로 띄어쓰기는 주어지지 않는다.

#### Output

해독된 문장과 주어진 암호문을 해독한 문장이 일치한다면 "y" 아니라면 "n"을 출력한다.

#### Sample Input 1

```
11
44 0 38 41 38 31 23 8 41 30 38
Hello World
```

#### Output for the Sample Input 1

```
y
```

#### Sample Input 2

```
5
12 3 34 52 0
apple
```

#### Output for the Sample Input 2

```
n
```

# Problem E

## 내접사각형 만들기

Time Limit: 1 Second

어떤 사각형의 네 꼭짓점이 원과 접할 때 이 사각형이 원에 내접한다고 한다. 세 선분이 주어지고 그중 가장 긴 선분이 내접할 원의 지름과 같을 때, 나머지 한 선분의 길이를 구하여라.

### Input

첫 번째 줄에 세 변의 길이를 나타내는 세 정수  $a, b, c$ 가 주어진다. ( $1 \leq a, b, c \leq 100$ )  
 세 선분은 내림차순으로 주어진다.

### Output

첫째 줄에 나머지 한 변의 길이를 출력한다.  
 원에 내접하는 사각형을 만들 수 없다면 -1을 출력한다.  
 단, 정답이 실수로 나오는 케이스는 주어지지 않는다.

#### Sample Input 1

2 1 1

#### Output for the Sample Input 1

1

#### Sample Input 2

1 1 1

#### Output for the Sample Input 2

-1

#### Sample Input 3

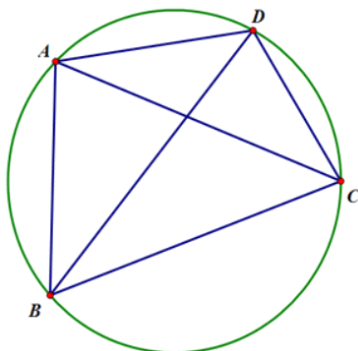
72 42 23

#### Output for the Sample Input 3

42

### Hint

톨레미의 정리에 따르면 다음과 같이 원에 내접한 사각형은 마주 보는 두 쌍의 변의 곱의 합이 두 대각선의 곱과 같다.



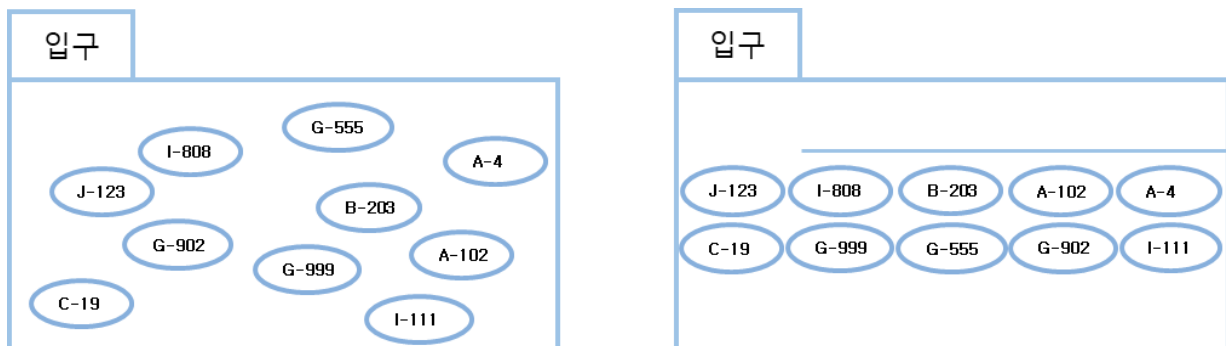
$$\overline{AB} \cdot \overline{CD} + \overline{AD} \cdot \overline{BC} = \overline{AC} \cdot \overline{BD}$$

# Problem F

## 줄서기

Time Limit: 1 Second

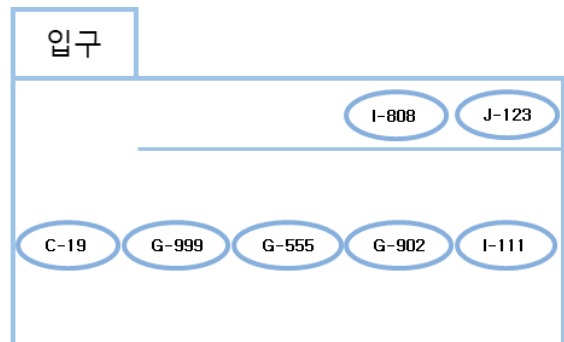
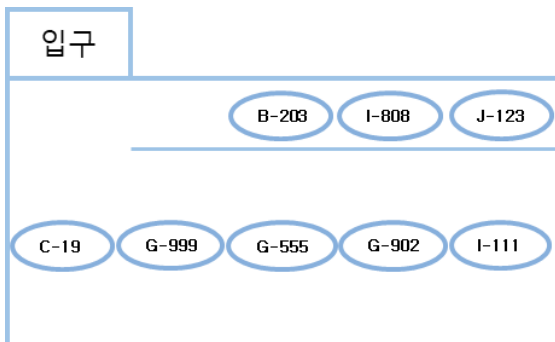
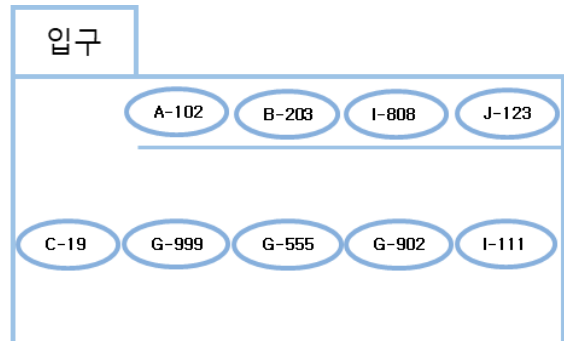
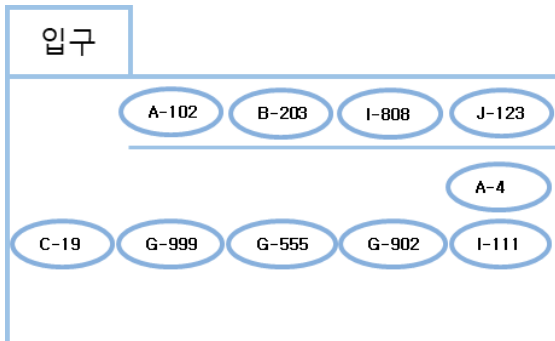
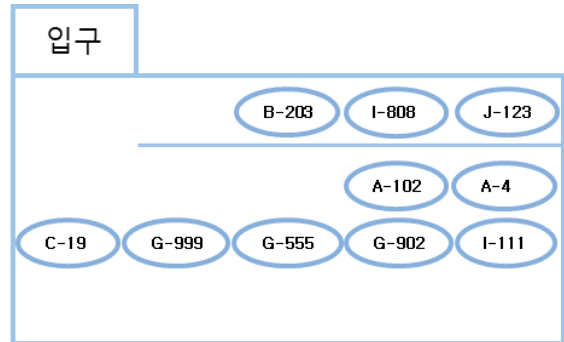
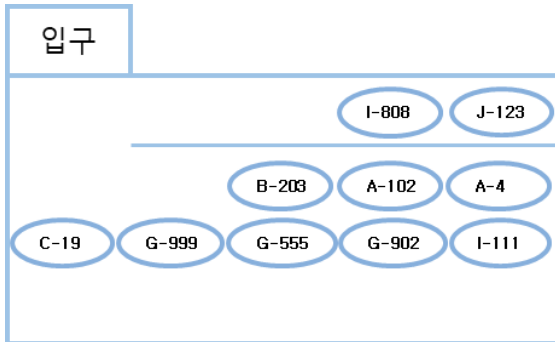
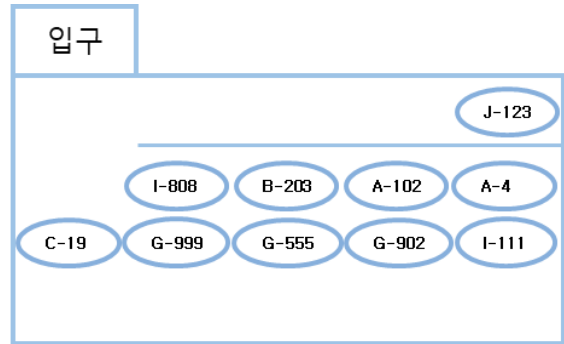
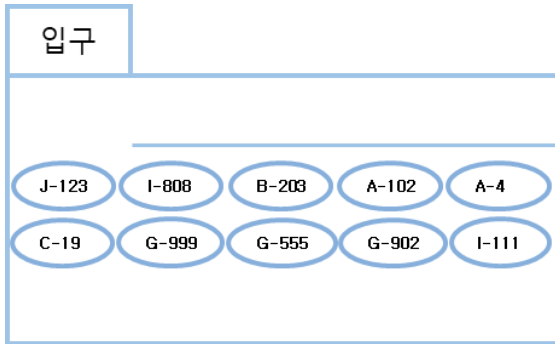
아이즈원의 팬인 시온이는 드디어 티켓팅에 성공하여 콘서트를 갔다. 콘서트장에 일찍 도착한 시온이는 기대하며 입장을 위해 줄을 섰다. 하지만 아이즈원의 인기대로 시온이를 포함한 많은 팬이 줄을 서고 있다. 콘서트의 입장을 시작했고 입장은 티켓 번호 순서대로 이루어졌다. 하지만 입구에 너무 많은 팬이 아무도 이동할 수 없는 상황이 되었고 주최 측에서 인원을 정렬시켜 다음과 같이 간신히 사람 한 줄이 설 수 있는 대기 공간을 만들었다.



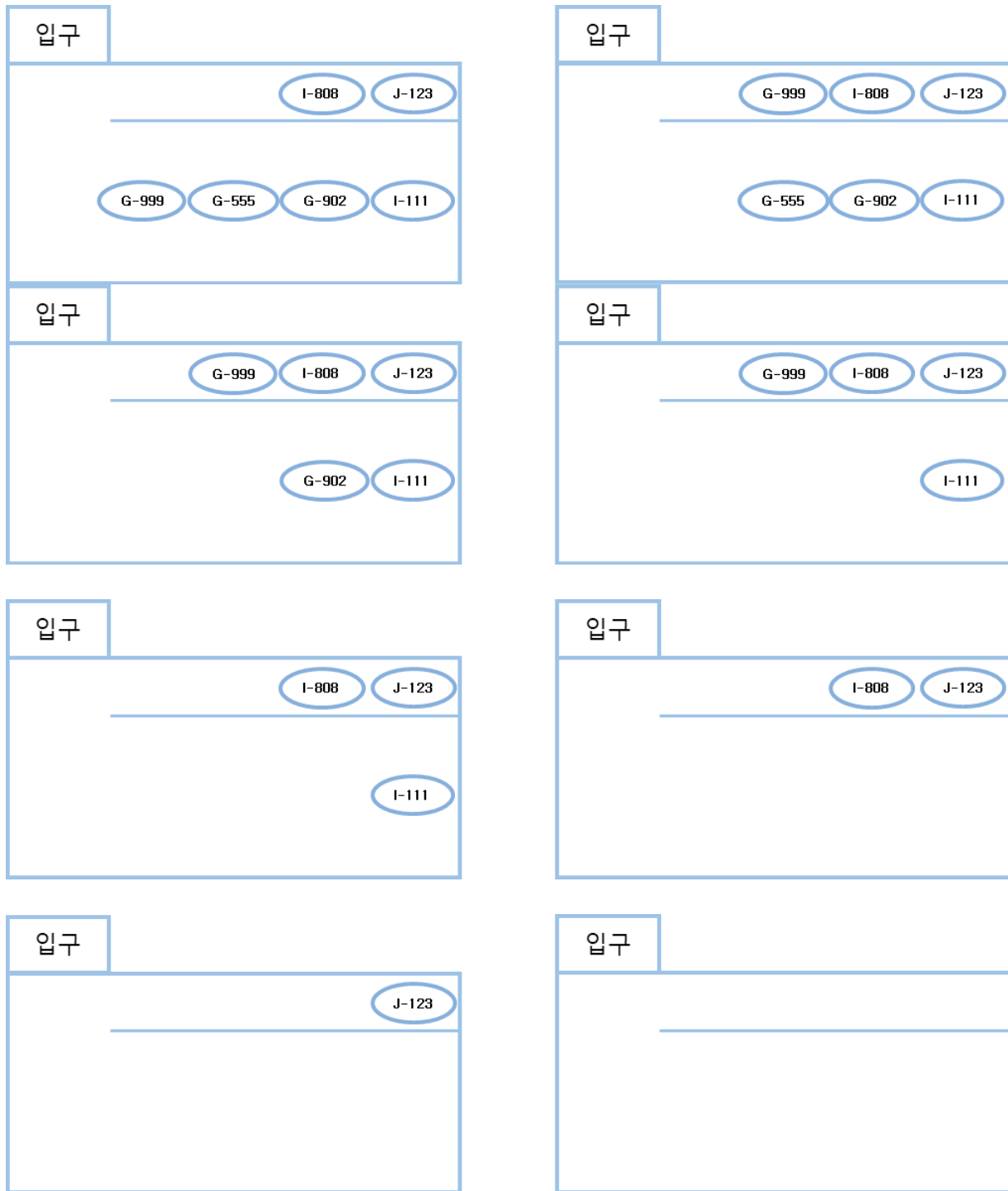
주최 측은 번호표 순서대로만 통과할 수 있는 입구를 만들어 두었지만, 줄에서는 마구잡이로 사람들이 기다리고 있다. 대기 공간을 이용하여 입장이 원활히 이루어지도록 하려고 한다. 콘서트장에 사람들이 제대로 들어갈 수 있는지 확인해보자.

사람들은 현재 5명씩 N 줄을 서 있고, 첫 번째 줄 맨 앞사람만 이동이 가능하다. 이 사람은 콘서트장으로 입장할 수도 있고 대기 공간에서 다시 기다릴 수도 있다. 한 줄의 사람이 다 이동했다면 그 다음 줄의 사람들이 이동한다. 대기 공간에서는 한 줄로만 설 수 있는 공간이 있으며, 대기 공간은 마지막에 들어온 사람부터 나갈 수 있다. 또 대기 공간에서 다시 줄로 돌아갈 수는 없다. 티켓은 A-123과 같이 한 개의 대문자 알파벳과 '-', 1000 미만의 숫자의 조합으로 이루어져 있다. 만약 숫자가 7이라면 A-7과 같이 주어진다. 티켓의 순서는 알파벳이 빠른 티켓이 빠르며, 동일하다면 더 작은 숫자가 더 빠르다. 티켓 번호는 중복되지 않는다.

위의 예시를 예로 들면 다음과 같이 모든 사람들이 입장할 수 있다. 대기 공간이 문제에서는 5명 이상도 일렬로 서있을 수 있다는 것에 주의하여야 한다.







**Input**

첫째 줄에는 줄에서 기다리고 있는 사람들의 줄 수  $N$ 이 주어진다. ( $1 \leq N \leq 100$ )  
 둘째 줄부터  $N$  개의 줄에는 한 줄에 서 있는 5명의 티켓 번호가 주어진다.  
 사람들이 서 있는 순서대로 입력이 주어진다.

**Output**

모든 사람이 무사히 콘서트장에 입장할 수 있다면 "GOOD"을 출력하고 그렇지 않다면 "BAD"를 출력한다.

**Sample Input 1****Output for the Sample Input 1**

1 G-555 B-203 A-102 A-504 C-719	GOOD
------------------------------------	------

**Sample Input 2****Output for the Sample Input 2**

2 J-123 I-808 B-203 A-102 A-4 C-19 G-999 G-555 G-902 I-111	GOOD
--	------

**Sample Input 3****Output for the Sample Input 3**

2 J-123 I-808 B-203 A-102 A-4 C-19 G-999 G-555 I-111 G-902	BAD
--	-----

## Problem G 케이크 자르기

Time Limit: 1 Second

생일을 맞이한 주성이가 생일 파티를 준비하려고 한다. 주성이는 일반 케이크 대신 평소 좋아하던 롤케이크를 준비했다. 롤 케이크에는 장식이 존재하여 일정한 크기로 자를 수 없다. 그래서 주성이는 미리 롤 케이크를 친구들의 수대로 잘라서 가장 작은 조각의 크기를 미리 알아보고 싶다. 하지만 짓곳은 주성이의 친구들은 생일파티에 몇 명이 참석하는지 직접적으로 알려주지를 않는다. 주성이가 롤 케이크를 몇 번 잘라야 하는지 리스트로 알려주기로 했다. 이 리스트에 있는 횟수대로 롤 케이크를 자를 때 가장 작은 롤 케이크의 길이가 가장 크게 되는 길이를 구해보자.

예를 들어 70cm의 롤 케이크에 자를 수 있는 지점이 5군데(10cm, 20cm, 35cm, 55cm, 60cm)가 있다고 하자. 만약 리스트의 숫자 중 하나가 3이라면 20cm, 35cm, 55cm 지점을 자를 때, 가장 작은 조각의 길이는 15cm로 최대가 된다.

### Input

첫 번째 줄에 자르는 횟수가 담긴 리스트의 길이  $N$  와 자를 수 있는 지점의 개수  $M$ , 그리고 롤 케이크의 길이인 정수  $L$ 이 주어진다. ( $1 \leq N \leq M \leq 1,000$ ,  $1 < L \leq 4,000,000$ )

다음  $M$  줄에 걸쳐 자를 수 있는 지점을 나타내는 정수  $S_i$ 가 주어진다. ( $1 \leq S_i < L$ )

다음  $N$  줄에 걸쳐 자르는 횟수를 나타내는 정수  $Q_i$ 가 주어진다. ( $1 \leq Q_i \leq M$ )

$S_i$ 와  $Q_i$ 는 중복된 지점은 주어지지 않으며 오름차순으로 주어진다.

### Output

$N$ 개 줄에 걸쳐 각 리스트에 있는 횟수대로 롤 케이크를 잘랐을 때 가장 작은 조각의 길이가 최대가 되는 길이를 출력한다.

#### Sample Input 1

```
2 5 70
10
20
35
55
60
3
4
```

#### Output for the Sample Input 1

```
15
10
```

## Problem H 문자열 비교하기

Time Limit: 1 Second

두 문자열을 비교하는 방법이 다음과 같은 두 가지 규칙을 따를 때, 두 문자열 간 차이의 최솟값이 얼마가 되는지를 구해보려 한다. 첫 번째로 각 문자 간의 차이는 알파벳 순서의 차이의 절댓값과 같다. 예를 들어, a는 첫 번째 알파벳이고 c는 세 번째 알파벳이므로, a와 c의 차이는  $|1-3| = 2$ 이다. 마찬가지로 a와 z의 차이는  $|1 - 26| = 25$ 이다. 두 번째 규칙은 두 문자열의 각 알파벳을 늘이는 것이 가능하다는 것이다. 위 두 규칙을 이용해 두 문자열의 문자 간 차이의 합을 구하면 문자열의 차이가 된다. 예를 들어 apple과 aple이 주어질 때, aple의 p를 늘려 apple로 만들 수 있다. 이 경우 두 문자열의 차이는 0이다. 임의의 두 문자열이 주어질 때, 두 문자열의 차이가 최소가 되는 값을 구해라.

단, 두 문자열에서 문자 간의 차이의 합을 구할 때는 반드시 두 문자열의 길이를 동일하게 만들어서 계산해야 한다.

### Input

첫 번째 줄에는 2개의 문자열의 길이인  $N$ 와  $M$ 가 주어진다. ( $1 \leq N, M \leq 300$ )

두 번째 줄에는 첫 번째 문자열이 주어진다.

세 번째 줄에는 두 번째 문자열이 주어진다.

단, 문자열에는 a~z까지의 알파벳 소문자만 주어진다.

### Output

두 문자열 간 차이의 최솟값을 구한다.

#### Sample Input 1

```
5 4  
apple  
apla
```

#### Output for the Sample Input 1

```
4
```

#### Sample Input 2

```
14 12  
aaaappppppplle  
applllllllee
```

#### Output for the Sample Input 2

```
0
```

# Problem I

## 나랏말싸미 America와 different~

Time Limit: 1 Second

토종 한국인 제임스는 외국에서 오래 살아 한글이 약한 김옥순을 위해 한글을 알려주려 한다. 외국에서 살아 받침을 잘 알지 못하는 김옥순을 위해 제임스는 한글의 받침에 대해 공부할 수 있는 게임을 만들었다. 게임은  $N \times M$ 의 표가 주어질 때 (1, 1)좌표에서 출발하여 (N, M) 좌표로 이동했을 때 가장 짧은 글자 수를 찾는 게임이다. 표에서 이동은 상하좌우 4방향으로 가능하다.

ㅏ	ㅓ	ㄴ	ㅗ
ㅓ	ㅇ	ㄴ	ㄷ
ㅁ	ㅓ	ㅓ	ㅓ
ㅓ	ㅓ	ㅁ	ㅓ
ㅁ	ㅓ	ㄷ	ㅓ

$N$ 이 5이고  $M$ 이 4인 위의 예시를 보면 "추매보대", "추우보대" 같은 단어를 만들어 낼 수 있지만 가장 짧은 글자 수는 3개로 총남대가 만들어질 수 있다. 김옥순 씨에게 문제를 내기 위해 게임의 정답이 무엇인지 알아야 하기 때문에 문제를 풀어보아야 한다. 한글 입출력을 원활하게 진행하기 위해 다음 표와 같이 자음과 모음 대신 숫자가 주어진다. 제임스를 도와 게임의 정답을 알아보자.

번호	0	1	2	3	4	5	6	7	8	9	10	11	12	13							
자음	ㄱ	ㄴ	ㄷ	ㄹ	ㅁ	ㅂ	ㅅ	ㅇ	ㅈ	ㅊ	ㅋ	ㆁ	ㅌ	ㅎ							
번호	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34
모음	ㅏ	ㅑ	ㅓ	ㅕ	ㅗ	ㅛ	ㅜ	ㅠ	ㅡ	ㅝ	ㅞ	ㅟ	ㅠ	ㅡ	ㅢ	ㅣ	ㅤ	ㅥ	ㅦ	ㅧ	ㅨ

글자를 만들 때 자음은 쌍자음이나 겹받침으로 이어 붙여서 사용하지 못하며, 문제에서 주어지는 모든 자음은 모든 모음에 대해 초성과 종성으로 사용할 수 있다고 가정한다.

### Input

첫 번째 줄에는 표의 가로와 세로 길이인  $N, M$ 이 주어진다. ( $4 \leq N \leq 50, 4 \leq M \leq 50$ )  
 두 번째 줄부터  $N$  줄에 걸쳐 표가 한 행씩 입력된다. 표의 각 숫자는 0보다 크거나 같고 34보다 작거나 같다.

### Output

표를 통해 만들어지는 가장 짧은 글자 수를 출력한다.  
 ( $N, M$ ) 좌표까지 갈 때 온전한 글자를 만들지 못할 경우 "BAD"를 출력한다.

**Sample Input 1****Output for the Sample Input 1**

5 4 9 27 1 6 27 7 1 2 4 27 14 16 15 5 4 20 4 22 2 15	3
---	---

한글로 입력이 주어졌다면 아래와 입력이 같다.

5 4

ㅅ ㅍ ㄴ ㅅ

ㅍ ㅇ ㄴ ㄷ

ㄱ ㅍ ㅅ ㅅ

ㅅ ㅅ ㄱ ㄷ

ㄱ ㅅ ㄷ ㅅ

정답은 "충남대" 가 된다.

**Sample Input 2****Output for the Sample Input 2**

5 4 9 27 1 6 27 7 1 2 4 27 14 16 15 5 4 2 4 22 2 2	BAD
---	-----

한글로 입력이 주어졌다면 아래와 입력이 같다.

5 4

ㅅ ㅍ ㄴ ㅅ

ㅍ ㅇ ㄴ ㄷ

ㄱ ㅍ ㅅ ㅅ

ㅅ ㅅ ㄱ ㄷ

ㄱ ㅅ ㄷ ㄷ

단어를 만들 수 없으므로 "BAD"가 된다.

## Problem J 우주 탐사선

Time Limit: 1 Second

우주 탐사선 ana호는 어떤 행성계를 탐사하기 위해 발사된다. 모든 행성을 탐사하는데 걸리는 최소 시간을 계산하려 한다. 입력으로는 ana호가 탐색할 행성의 개수와 ana호가 발사되는 행성의 위치와 ana호가 행성 간 이동을 하는데 걸리는 시간이 2차원 행렬로 주어진다. 행성의 위치는 0부터 시작하여 0은 행렬에서 0번째 인덱스에 해당하는 행성을 의미한다. 2차원 행렬에서  $i, j$  번 요소는  $i$  번째 행성에서  $j$  번째 행성에 도달하는데 걸리는 시간을 나타낸다.  $i$ 와  $j$ 가 같을 때는 항상 0이 주어진다. 모든 행성을 탐사하는데 걸리는 최소 시간을 계산하여라.

탐사 후 다시 시작 행성으로 돌아올 필요는 없으며 이미 방문한 행성도 중복해서 갈 수 있다.

### Input

첫 번째 줄에는 행성의 개수  $N$ 과 ana호가 발사되는 행성의 위치  $K$ 가 주어진다.

$(2 \leq N \leq 10, 0 \leq K < N)$

다음  $N$  줄에 걸쳐 각 행성 간 이동 시간  $T_{ij}$ 가  $N$  개 씩 띄어쓰기로 구분되어 주어진다.

$(0 \leq T_{ij} \leq 1000)$

### Output

모든 행성을 탐사하기 위한 최소 시간을 출력한다.

#### Sample Input 1

```
3 0
0 30 1
1 0 29
28 1 0
```

#### Output for the Sample Input 1

```
2
```

#### Sample Input 2

```
4 1
0 83 38 7
15 0 30 83
67 99 0 44
14 46 81 0
```

#### Output for the Sample Input 2

```
74
```