

# 코딩 테스트 모의고사 (C형)

## < 문제지 >

|       |     |         |     |
|-------|-----|---------|-----|
| 난이도   | 상   | 문제 풀이시간 | 3시간 |
| 문제 개수 | 3문제 | 합격 커트라인 | 2문제 |

코딩 테스트 모의고사 문제는 인터넷 검색이 불가능한 상황에서 제한 시간안에 풀어주세요. 소스코드 작성 및 프로그램 실행 결과를 계산할 수 있는 Python 3.7 개발 환경은 제공된다고 가정합니다.

## 문제 1. 시각

(시간 제한: 1초, 메모리 제한: 256MB)

정수  $N$  과  $K$  가 입력되었을 때 00시 00분 00초부터  $N$ 시 59분 59초까지의 모든 시각 중에서  $K$ 가 하나라도 포함되는 모든 시각을 세는 프로그램을 작성하시오. 시각을 셀 때는 디지털 시계를 기준으로, 초 단위로만 시각을 구분한다.

예를 들어  $K = 3$  일 때, 다음의 시각들은 3이 하나 이상 포함되어 있으므로 세어야 하는 시각의 대표적인 예시이다.

- ① 23시 00분 00초
- ② 07시 08분 33초

반면에 다음의 시각들은 3이 하나도 포함되어 있지 않으므로 세면 안 되는 예시이다.

- ① 15시 02분 55초
- ② 18시 27분 45초

## 입력 조건

첫째 줄에 정수  $N$  과  $K$  가 공백을 기준으로 구분되어 주어진다. ( $0 \leq N \leq 23, 0 \leq K \leq 9$ )

## 출력 조건

00 시 00 분 00 초부터  $N$  시 59 분 59 초까지의 모든 시각 중에서  $K$  가 하나라도 포함되는 시각들의 수를 출력한다.

## 입력 예시

5 3

## 출력 예시

11475

## 문제 2. 함께 블록 쌓기

(시간 제한: 1초, 메모리 제한: 256MB)

1 번부터  $N$  번까지의 학생들은 각각 블록들을 가지고 있다. 학생마다 최대  $M$  개의 블록을 가지고 있을 수 있으며, 한 명의 학생이 가지고 있는 모든 블록들의 높이는 서로 다르다. 이 때 1 번부터  $N$  번까지의 학생들이 가진 블록을 차례대로 사용하여 바닥에서부터 쌓아올려 하나의 탑을 만들고자 한다.

단, 어떤 학생의 블록은 사용하지 않아도 되며 한 학생당 최대 1 개의 블록만을 사용할 수 있다.

1 번부터  $N$  번까지의 학생들이 가지고 있는 블록들에 대한 정보가 주어졌을 때, 높이가 정확히  $H$  인 탑을 만들 수 있는 경우의 수를 계산하는 프로그램을 작성하시오.

예를 들어  $N = 3$ ,  $M = 3$ ,  $H = 5$  일 때, 각 학생마다 가지고 있는 블록들의 높이가 다음과 같다고 가정하자.

1번 학생: 2, 3, 5

2번 학생: 3, 5

3번 학생: 1, 2, 3

이 때, 탑의 높이가 정확히 5가 되도록 블록을 쌓는 경우로는 다음의 6가지가 존재한다. (블록을 사용하지 않는 경우는  $X$ 로 표시하였다.)

|       |       |       |
|-------|-------|-------|
| 1번 학생 | 2번 학생 | 3번 학생 |
|-------|-------|-------|

|   |   |   |
|---|---|---|
| X | 3 | 2 |
| X | 5 | X |
| 2 | X | 3 |
| 2 | 3 | X |
| 3 | X | 2 |
| 5 | X | X |

### 입력 조건

첫째 줄에 자연수  $N, M, H$ 가 공백을 기준으로 구분되어 주어진다.  
( $1 \leq N \leq 50, 1 \leq M \leq 10, 1 \leq H \leq 1,000$ ) 둘째 줄부터  $N$ 개의 줄에 걸쳐서 각 학생이 가진 블록들의 높이가 공백을 기준으로 구분되어 주어진다.  
단, 모든 블록의 높이는 1,000 이하의 자연수이며 한 명의 학생이 가지고 있는 모든 블록들의 높이는 서로 다르게 주어진다.

### 출력 조건

첫째 줄에 높이가  $H$ 인 탑을 만드는 경우의 수를 10,007로 나눈 나머지를 출력한다.

### 입력 예시

3 3 5  
2 3 5  
3 5  
1 2 3

### 출력 예시

6

## 문제 3. 감시 피하기

(시간 제한: 2초, 메모리 제한: 256MB)

$N \times N$  크기의 복도가 있다. 복도는  $1 \times 1$  크기의 칸으로 나누어지며, 특정한 위치에는 선생님, 학생, 혹은 장애물이 위치할 수 있다. 현재 몇 명의 학생들은 수업시간에 몰래 복도로 빠져나왔는데, 복도로 빠져나온 학생들은 선생님의 감시에 들키지 않는 것이 목표이다.

각 선생님들은 자신의 위치에서 상, 하, 좌, 우 4가지 방향으로 감시를 진행한다. 단 복도에 장애물이 위치한 경우, 선생님은 장애물 뒤편에 숨어 있는 학생들은 볼 수 없다. 또한 선생님은 상, 하, 좌, 우 4가지 방향에 대하여, 아무리 멀리 있더라도 장애물로 막히기 전까지의 학생들은 모두 볼 수 있다고 가정하자.

다음과 같이  $3 \times 3$  크기의 복도의 정보가 주어진 상황을 확인해보자. 단, 본 문제에서 위치 값을 나타낼 때는 (행, 열)의 형태로 표현한다. 선생님이 존재하는 칸은  $T$ , 학생이 존재하는 칸은  $S$ , 장애물이 존재하는 칸은  $O$ 로 표시하였다. 아래 그림과 같이  $(3, 1)$ 의 위치에는 선생님이 존재하며  $(1, 1), (2, 1), (3, 3)$ 의 위치에는 학생이 존재한다. 그리고  $(1, 2), (2, 2), (3, 2)$ 의 위치에는 장애물이 존재한다.

|   |   |   |
|---|---|---|
| S | O |   |
| S | O |   |
| T | O | S |

이 때  $(3, 3)$ 의 위치에 존재하는 학생은 장애물 뒤편에 숨어 있기 때문에 감시를 피할 수 있다. 하지만  $(1, 1)$ 과  $(2, 1)$ 의 위치에 존재하는 학생은 선생님에게 들키게 된다.

학생들은 복도의 빈 칸 중에서 장애물을 설치할 위치를 골라, 정확히 3개의 장애물을 설치해야 한다. 결과적으로 3개의 장애물을 설치하여 모든 학생들을 감시로부터 피하도록 할 수 있는지 계산하고자 한다.  $N \times N$  크기의 복도에서 학생 및 선생님의 위치 정보가 주어졌을 때, 장애물을 정확히 3개 설치하는 모든 경우 중에서, 모든 학생들이 선생님의 감시로부터 피할 수 있도록 하는 경우가 존재하는지 출력하는 프로그램을 작성하시오.

예를 들어  $N = 5$  일 때, 다음과 같이 선생님 및 학생의 위치 정보가 주어졌다고 가정하자.

|   |   |   |  |   |
|---|---|---|--|---|
|   | S |   |  | T |
| T |   | S |  |   |
|   |   |   |  |   |
|   | T |   |  |   |

|  |  |   |  |  |
|--|--|---|--|--|
|  |  | T |  |  |
|--|--|---|--|--|

이 때 다음과 같이 3개의 장애물을 설치하면, 모든 학생들을 선생님의 감시로부터 피하도록 할 수 있다.

|   |   |   |   |   |
|---|---|---|---|---|
|   | S |   | O | T |
| T | O | S |   |   |
|   |   | O |   |   |
|   | T |   |   |   |
|   |   | T |   |   |

### 입력 조건

첫째 줄에 자연수  $N$ 이 주어진다. ( $3 \leq N \leq 6$ ) 둘째 줄에  $N$ 개의 줄에 걸쳐서 복도의 정보가 주어진다. 각 행에서는  $N$ 개의 원소가 공백을 기준으로 구분되어 주어진다. 해당 위치에 학생이 있다면  $S$ , 선생님이 있다면  $T$ , 아무것도 존재하지 않는다면  $X$ 가 주어진다.

단, 전체 선생님의 수는 5이하의 자연수, 전체 학생의 수는 30 이하의 자연수이며 항상 빈 칸의 개수는 3개 이상이다.

### 출력 조건

첫째 줄에 정확히 3개의 장애물을 설치하여 모든 학생들을 감시로부터 피하도록 할 수 있는지의 여부를 출력한다. 모든 학생들을 감시로부터 피하도록 할 수 있다면  $YES$ , 그렇지 않다면  $NO$ 를 출력한다.

### 입력 예시 1

```

5
XSXXT
TXSXX
XXXXX
XTXXX
XXTXX

```

### 출력 예시 1

YES

## 입력 예시 2

4

S S S T

X X X X

X X X X

T T T X

## 출력 예시 2

NO