

취업 준비 스터디 6주차 멘토링

2022.11.02



발표자 : 박동학

1. About Plan

2. 이번주 문제 해설

3. 라이브 코딩

4. 면접 준비 (운영체제, 개발 상식)

5. QnA

6. 삼성 면접 Tip (면접 보시는 분들 대상)

1주 - 취업개요

What For Next ?

2주 - 코딩테스트 개요

3주 - 코딩테스트

Coding Test + Computer Science

4주 - 클린 코딩

5주 - 코딩테스트

→ Class 5

6주 - 코딩테스트

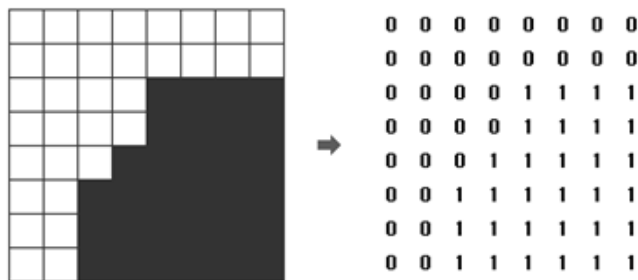
→ <https://gyoogle.dev/blog/>

문제

흑백 영상을 압축하여 표현하는 데이터 구조로 쿼드 트리(Quad Tree)라는 방법이 있다. 흰 점을 나타내는 0과 검은 점을 나타내는 1로만 이루어진 영상(2차원 배열)에서 같은 숫자의 점들이 한 곳에 많이 몰려있으면, 쿼드 트리에서는 이를 압축하여 간단히 표현할 수 있다.

주어진 영상이 모두 0으로만 되어 있으면 압축 결과는 "0"이 되고, 모두 1로만 되어 있으면 압축 결과는 "1"이 된다. 만약 0과 1이 섞여 있으면 전체를 한 번에 나타내지를 못하고, 왼쪽 위, 오른쪽 위, 왼쪽 아래, 오른쪽 아래, 이렇게 4개의 영상으로 나누어 압축하게 되며, 이 4개의 영역을 압축한 결과를 차례대로 괄호 안에 묶어서 표현한다

분할 정복 & 재귀



위 그림에서 왼쪽의 영상은 오른쪽의 배열과 같이 숫자로 주어지며, 이 영상을 쿼드 트리 구조를 이용하여 압축하면 "(0(0011)(0(0111)01)1)"로 표현된다. $N \times N$ 크기의 영상이 주어질 때, 이 영상을 압축한 결과를 출력하는 프로그램을 작성하시오.

입력

첫째 줄에는 영상의 크기를 나타내는 숫자 N 이 주어진다. N 은 언제나 2의 제곱수로 주어지며, $1 \leq N \leq 64$ 의 범위를 가진다. 두 번째 줄부터는 길이 N 의 문자열이 N 개 들어온다. 각 문자열은 0 또는 1의 숫자로 이루어져 있으며, 영상의 각 점들을 나타낸다.

출력

영상을 압축한 결과를 출력한다.

문제

N×M크기의 배열로 표현되는 미로가 있다.

1	0	1	1	1	1
1	0	1	0	1	0
1	0	1	0	1	1
1	1	1	0	1	1

단순 탐색 (DFS&BFS)

미로에서 1은 이동할 수 있는 칸을 나타내고, 0은 이동할 수 없는 칸을 나타낸다. 이러한 미로가 주어졌을 때, (1, 1)에서 출발하여 (N, M)의 위치로 이동할 때 지나야 하는 최소의 칸 수를 구하는 프로그램을 작성하시오. 한 칸에서 다른 칸으로 이동할 때, 서로 인접한 칸으로만 이동할 수 있다.

위의 예에서는 15칸을 지나야 (N, M)의 위치로 이동할 수 있다. 칸을 셀 때에는 시작 위치와 도착 위치도 포함한다.

입력

첫째 줄에 두 정수 N, M($2 \leq N, M \leq 100$)이 주어진다. 다음 N개의 줄에는 M개의 정수로 미로가 주어진다. 각각의 수들은 붙어서 입력으로 주어진다.

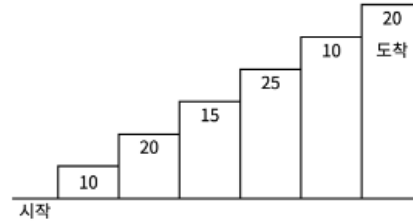
출력

첫째 줄에 지나야 하는 최소의 칸 수를 출력한다. 항상 도착위치로 이동할 수 있는 경우만 입력으로 주어진다.

3. 계단 오르기

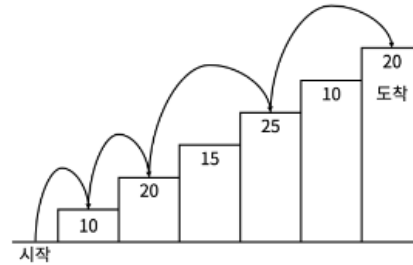
문제

계단 오르기 게임은 계단 아래 시작점부터 계단 꼭대기에 위치한 도착점까지 가는 게임이다. <그림 1>과 같이 각각의 계단에는 일정한 점수가 쓰여 있는데 계단을 밟으면 그 계단에 쓰여 있는 점수를 얻게 된다.



<그림 1>

예를 들어 <그림 2>와 같이 시작점에서부터 첫 번째, 두 번째, 네 번째, 여섯 번째 계단을 밟아 도착점에 도달하면 총 점수는 $10 + 20 + 25 + 20 = 75$ 점이 된다.



<그림 2>

계단 오르는 데는 다음과 같은 규칙이 있다.

1. 계단은 한 번에 한 계단씩 또는 두 계단씩 오를 수 있다. 즉, 한 계단을 밟으면서 이어서 다음 계단이나, 다음 다음 계단으로 오를 수 있다.
2. 연속된 세 개의 계단을 모두 밟아서는 안 된다. 단, 시작점은 계단에 포함되지 않는다.
3. 마지막 도착 계단은 반드시 밟아야 한다.

따라서 첫 번째 계단을 밟고 이어 두 번째 계단이나, 세 번째 계단으로 오를 수 있다. 하지만, 첫 번째 계단을 밟고 이어 네 번째 계단으로 올라가거나, 첫 번째, 두 번째, 세 번째 계단을 연속해서 모두 밟을 수는 없다.

각 계단에 쓰여 있는 점수가 주어질 때 이 게임에서 얻을 수 있는 총 점수의 최댓값을 구하는 프로그램을 작성하시오.

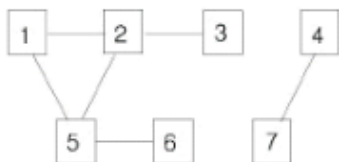
DP

4. 바이러스

문제

신종 바이러스인 웜 바이러스는 네트워크를 통해 전파된다. 한 컴퓨터가 웜 바이러스에 걸리면 그 컴퓨터와 네트워크 상에서 연결되어 있는 모든 컴퓨터는 웜 바이러스에 걸리게 된다.

예를 들어 7대의 컴퓨터가 <그림 1>과 같이 네트워크 상에서 연결되어 있다고 하자. 1번 컴퓨터가 웜 바이러스에 걸리면 웜 바이러스는 2번과 5번 컴퓨터를 거쳐 3번과 6번 컴퓨터까지 전파되어 2, 3, 5, 6 네 대의 컴퓨터는 웜 바이러스에 걸리게 된다. 하지만 4번과 7번 컴퓨터는 1번 컴퓨터와 네트워크 상에서 연결되어 있지 않기 때문에 영향을 받지 않는다.



< 그림 1 >

Union Find

어느 날 1번 컴퓨터가 웜 바이러스에 걸렸다. 컴퓨터의 수와 네트워크 상에서 서로 연결되어 있는 정보가 주어질 때, 1번 컴퓨터를 통해 웜 바이러스에 걸리게 되는 컴퓨터의 수를 출력하는 프로그램을 작성하시오.

입력

첫째 줄에는 컴퓨터의 수가 주어진다. 컴퓨터의 수는 100 이하이고 각 컴퓨터에는 1번 부터 차례대로 번호가 매겨진다. 둘째 줄에는 네트워크 상에서 직접 연결되어 있는 컴퓨터 쌍의 수가 주어진다. 이어서 그 수만큼 한 줄에 한 쌍씩 네트워크 상에서 직접 연결되어 있는 컴퓨터의 번호 쌍이 주어진다.

출력

1번 컴퓨터가 웜 바이러스에 걸렸을 때, 1번 컴퓨터를 통해 웜 바이러스에 걸리게 되는 컴퓨터의 수를 첫째 줄에 출력한다.

5. 색종이 만들기

아래 <그림 1>과 같이 여러개의 정사각형칸들로 이루어진 정사각형 모양의 종이가 주어져 있고, 각 정사각형들은 하얀색으로 칠해져 있거나 파란색으로 칠해져 있다. 주어진 종이를 일정한 규칙에 따라 잘라서 다양한 크기를 가진 정사각형 모양의 하얀색 또는 파란색 색종이를 만들려고 한다.

1	1	0	0	0	0	1	1
1	1	0	0	0	0	1	1
0	0	0	0	1	1	0	0
0	0	0	0	1	1	0	0
1	0	0	0	1	1	1	1
0	1	0	0	1	1	1	1
0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1

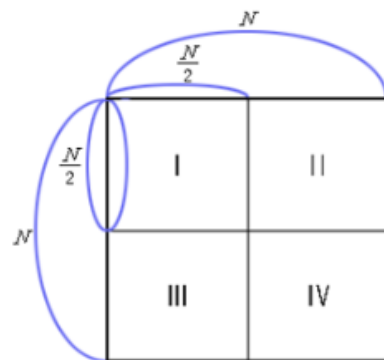
<그림 1> 8×8 종이

전체 종이의 크기가 $N \times N$ ($N=2^k$, k 는 1 이상 7 이하의 자연수) 이라면 종이를 자르는 규칙은 다음과 같다.

전체 종이가 모두 같은 색으로 칠해져 있지 않으면 가로와 세로로 중간 부분을 잘라서 <그림 2>의 I, II, III, IV와 같이 똑같은 크기의 네 개의 $N/2 \times N/2$ 색종이로 나눈다. 나누어진 종이 I, II, III, IV 각각에 대해서도 앞에서와 마찬가지로 모두 같은 색으로 칠해져 있지 않으면 같은 방법으로 똑같은 크기의 네 개의 색종이로 나눈다. 이와 같은 과정을 잘라진 종이가 모두 하얀색 또는 모두 파란색으로 칠해져 있거나, 하나의 정사각형 칸이 되어 더 이상 자를 수 없을 때까지 반복한다.

위와 같은 규칙에 따라 잘랐을 때 <그림 3>은 <그림 1>의 종이를 처음 나눈 후의 상태를, <그림 4>는 두 번째 나눈 후의 상태를, <그림 5>는 최종적으로 만들어진 다양한 크기의 9장의 하얀색 색종이와 7장의 파란색 색종이를 보여주고 있다.

분할 정복



<그림 2>

1	1	0	0	0	0	1	1
1	1	0	0	0	0	1	1
0	0	0	0	1	1	0	0
0	0	0	0	1	1	0	0
1	0	0	0	1	1	1	1
0	1	0	0	1	1	1	1
0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1

<그림 3> 처음 나눈 후의 상태

1	1	0	0	0	0	1	1
1	1	0	0	0	0	1	1
0	0	0	0	1	1	0	0
0	0	0	0	1	1	0	0
1	0	0	0	1	1	1	1
0	1	0	0	1	1	1	1
0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1

<그림 4> 두 번째 나눈 후의 상태

1	1	0	0	0	0	1	1
1	1	0	0	0	0	1	1
0	0	0	0	1	1	0	0
0	0	0	0	1	1	0	0
1	0	0	0	1	1	1	1
0	1	0	0	1	1	1	1
0	0	1	1	1	1	1	1
0	0	1	1	1	1	1	1

<그림 5> 최종적으로 나누어진 색종이들

입력으로 주어진 종이의 한 변의 길이 N 과 각 정사각형칸의 색(하얀색 또는 파란색)이 주어질 때 잘라진 하얀색 색종이와 파란색 색종이의 개수를 구하는 프로그램을 작성하시오.

입력

첫째 줄에는 전체 종이의 한 변의 길이 N 이 주어져 있다. N 은 2, 4, 8, 16, 32, 64, 128 중 하나이다. 색종이의 각 가로줄의 정사각형칸들의 색이 뒷줄부터 차례로 둘째 줄부터 마지막 줄까지 주어진다. 하얀색으로 칠해진 칸은 0, 파란색으로 칠해진 칸은 1로 주어지며, 각 숫자 사이에는 빈칸이 하나씩 있다.

출력

첫째 줄에는 잘라진 하얀색 색종이의 개수를 출력하고, 둘째 줄에는 파란색 색종이의 개수를 출력한다.

6. 단지번호 붙이기

문제

<그림 1>과 같이 정사각형 모양의 지도가 있다. 1은 집이 있는 곳을, 0은 집이 없는 곳을 나타낸다. 철수는 이 지도를 가지고 연결된 집의 모임인 단지를 정의하고, 단지에 번호를 붙이려 한다. 여기서 연결되었다는 것은 어떤 집이 좌우, 혹은 아래위로 다른 집이 있는 경우를 말한다. 대각선상에 집이 있는 경우는 연결된 것이 아니다. <그림 2>는 <그림 1>을 단지별로 번호를 붙인 것이다. 지도를 입력하여 단지수를 출력하고, 각 단지에 속하는 집의 수를 오름차순으로 정렬하여 출력하는 프로그램을 작성하시오.

탐색

0	1	1	0	1	0	0
0	1	1	0	1	0	1
1	1	1	0	1	0	1
0	0	0	0	1	1	1
0	1	0	0	0	0	0
0	1	1	1	1	1	0
0	1	1	1	0	0	0

<그림 1>

0	1	1	0	2	0	0
0	1	1	0	2	0	2
1	1	1	0	2	0	2
0	0	0	0	2	2	2
0	3	0	0	0	0	0
0	3	3	3	3	3	0
0	3	3	3	0	0	0

<그림 2>

입력

첫 번째 줄에는 지도의 크기 N (정사각형이므로 가로와 세로의 크기는 같으며 $5 \leq N \leq 25$)이 입력되고, 그 다음 N 줄에는 각각 N 개의 자료(0 혹은 1)가 입력된다.

출력

첫 번째 줄에는 총 단지수를 출력하시오. 그리고 각 단지내 집의 수를 오름차순으로 정렬하여 한 줄에 하나씩 출력하시오.

7. AC

AC

성공

다국어

☆

한국어 ▼

5 골드 V

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	256 MB	91817	20283	14309	19.835%

문제

선영이는 주말에 할 일이 없어서 새로운 언어 AC를 만들었다. AC는 정수 배열에 연산을 하기 위해 만든 언어이다. 이 언어에는 두 가지 함수 R(뒤집기)과 D(버리기)가 있다.

함수 R은 배열에 있는 수의 순서를 뒤집는 함수이고, D는 첫 번째 수를 버리는 함수이다. 배열이 비어있는데 D를 사용한 경우에는 에러가 발생한다.

함수는 조합해서 한 번에 사용할 수 있다. 예를 들어, "AB"는 A를 수행한 다음에 바로 이어서 B를 수행하는 함수이다. 예를 들어, "RDD"는 배열을 뒤집은 다음 처음 두 수를 버리는 함수이다.

배열의 초기값과 수행할 함수가 주어졌을 때, 최종 결과를 구하는 프로그램을 작성하시오.

입력

첫째 줄에 테스트 케이스의 개수 T가 주어진다. T는 최대 100이다.

각 테스트 케이스의 첫째 줄에는 수행할 함수 p가 주어진다. p의 길이는 1보다 크거나 같고, 100,000보다 작거나 같다.

다음 줄에는 배열에 들어있는 수의 개수 n이 주어진다. ($0 \leq n \leq 100,000$)

다음 줄에는 $[x_1, \dots, x_n]$ 과 같은 형태로 배열에 들어있는 정수가 주어진다. ($1 \leq x_i \leq 100$)

전체 테스트 케이스에 주어지는 p의 길이의 합과 n의 합은 70만을 넘지 않는다.

출력

각 테스트 케이스에 대해서, 입력으로 주어진 정수 배열에 함수를 수행한 결과를 출력한다. 만약, 에러가 발생한 경우에는 error를 출력한다.

그냥 구현!

8. IOIOI

IOIOI

성공

서브테스크

다국어

☆

한국어 ▾

1 실버 I

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	256 MB	21695	6460	5265	31.252%

문제

$N+1$ 개의 I 와 N 개의 o 로 이루어져 있으면, I 와 o 이 교대로 나오는 문자열을 P_N 이라고 한다.

- P_1 IOI
- P_2 IOIOI
- P_3 IOIOIOI
- P_N IOIOI...OI (o 가 N 개)

그냥 구현!

I 와 o 로만 이루어진 문자열 S 와 정수 N 이 주어졌을 때, S 안에 P_N 이 몇 군데 포함되어 있는지 구하는 프로그램을 작성하시오.

입력

첫째 줄에 N 이 주어진다. 둘째 줄에는 S 의 길이 M 이 주어지며, 셋째 줄에 S 가 주어진다.

출력

S 에 P_N 이 몇 군데 포함되어 있는지 출력한다.

제한

- $1 \leq N \leq 1,000,000$
- $2N+1 \leq M \leq 1,000,000$
- S 는 I 와 o 로만 이루어져 있다.

카잉 달력

성공

다국어

1 실버 I

수학 문제..

☆

한국어 ▾

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	256 MB	53751	13043	9725	25.522%

문제

최근에 ICPC 탐사대는 남아메리카의 잉카 제국이 놀라운 문명을 지닌 카잉 제국을 토대로 하여 세워졌다는 사실을 발견했다. 카잉 제국의 백성들은 특이한 달력을 사용한 것으로 알려져 있다. 그들은 M 과 N 보다 작거나 같은 두 개의 자연수 x, y 를 가지고 각 년도를 $\langle x:y \rangle$ 와 같은 형식으로 표현하였다. 그들은 이 세상의 시초에 해당하는 첫 번째 해를 $\langle 1:1 \rangle$ 로 표현하고, 두 번째 해를 $\langle 2:2 \rangle$ 로 표현하였다. $\langle x:y \rangle$ 의 다음 해를 표현한 것을 $\langle x':y' \rangle$ 이라고 하자. 만일 $x < M$ 이면 $x' = x + 1$ 이고, 그렇지 않으면 $x' = 1$ 이다. 같은 방식으로 만일 $y < N$ 이면 $y' = y + 1$ 이고, 그렇지 않으면 $y' = 1$ 이다. $\langle M:N \rangle$ 은 그들 달력의 마지막 해로서, 이 해에 세상의 종말이 도래한다는 예언이 전해 온다.

예를 들어, $M = 10$ 이고 $N = 12$ 라고 하자. 첫 번째 해는 $\langle 1:1 \rangle$ 로 표현되고, 11번째 해는 $\langle 1:11 \rangle$ 로 표현된다. $\langle 3:1 \rangle$ 은 13번째 해를 나타내고, $\langle 10:12 \rangle$ 는 마지막인 60번째 해를 나타낸다.

네 개의 정수 M, N, x 와 y 가 주어질 때, $\langle M:N \rangle$ 이 카잉 달력의 마지막 해라고 하면 $\langle x:y \rangle$ 는 몇 번째 해를 나타내는지 구하는 프로그램을 작성하라.

입력

입력 데이터는 표준 입력을 사용한다. 입력은 T 개의 테스트 데이터로 구성된다. 입력의 첫 번째 줄에는 입력 데이터의 수를 나타내는 정수 T 가 주어진다. 각 테스트 데이터는 한 줄로 구성된다. 각 줄에는 네 개의 정수 M, N, x 와 y 가 주어진다. ($1 \leq M, N \leq 40,000, 1 \leq x \leq M, 1 \leq y \leq N$) 여기서 $\langle M:N \rangle$ 은 카잉 달력의 마지막 해를 나타낸다.

출력

출력은 표준 출력을 사용한다. 각 테스트 데이터에 대해, 정수 k 를 한 줄에 출력한다. 여기서 k 는 $\langle x:y \rangle$ 가 k 번째 해를 나타내는 것을 의미한다. 만일 $\langle x:y \rangle$ 에 의해 표현되는 해가 없다면, 즉, $\langle x:y \rangle$ 가 유효하지 않은 표현이면, -1을 출력한다.

카잉 달력

성공

다국어

1 실버 I

수학 문제..

☆

한국어 ▾

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	256 MB	53751	13043	9725	25.522%

문제

최근에 ICPC 탐사대는 남아메리카의 잉카 제국이 놀라운 문명을 지닌 카잉 제국을 토대로 하여 세워졌다는 사실을 발견했다. 카잉 제국의 백성들은 특이한 달력을 사용한 것으로 알려져 있다. 그들은 M 과 N 보다 작거나 같은 두 개의 자연수 x, y 를 가지고 각 년도를 $\langle x:y \rangle$ 와 같은 형식으로 표현하였다. 그들은 이 세상의 시초에 해당하는 첫 번째 해를 $\langle 1:1 \rangle$ 로 표현하고, 두 번째 해를 $\langle 2:2 \rangle$ 로 표현하였다. $\langle x:y \rangle$ 의 다음 해를 표현한 것을 $\langle x':y' \rangle$ 이라고 하자. 만일 $x < M$ 이면 $x' = x + 1$ 이고, 그렇지 않으면 $x' = 1$ 이다. 같은 방식으로 만일 $y < N$ 이면 $y' = y + 1$ 이고, 그렇지 않으면 $y' = 1$ 이다. $\langle M:N \rangle$ 은 그들 달력의 마지막 해로서, 이 해에 세상의 종말이 도래한다는 예언이 전해 온다.

예를 들어, $M = 10$ 이고 $N = 12$ 라고 하자. 첫 번째 해는 $\langle 1:1 \rangle$ 로 표현되고, 11번째 해는 $\langle 1:11 \rangle$ 로 표현된다. $\langle 3:1 \rangle$ 은 13번째 해를 나타내고, $\langle 10:12 \rangle$ 는 마지막인 60번째 해를 나타낸다.

네 개의 정수 M, N, x 와 y 가 주어질 때, $\langle M:N \rangle$ 이 카잉 달력의 마지막 해라고 하면 $\langle x:y \rangle$ 는 몇 번째 해를 나타내는지 구하는 프로그램을 작성하라.

입력

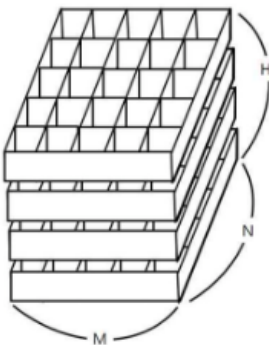
입력 데이터는 표준 입력을 사용한다. 입력은 T 개의 테스트 데이터로 구성된다. 입력의 첫 번째 줄에는 입력 데이터의 수를 나타내는 정수 T 가 주어진다. 각 테스트 데이터는 한 줄로 구성된다. 각 줄에는 네 개의 정수 M, N, x 와 y 가 주어진다. ($1 \leq M, N \leq 40,000, 1 \leq x \leq M, 1 \leq y \leq N$) 여기서 $\langle M:N \rangle$ 은 카잉 달력의 마지막 해를 나타낸다.

출력

출력은 표준 출력을 사용한다. 각 테스트 데이터에 대해, 정수 k 를 한 줄에 출력한다. 여기서 k 는 $\langle x:y \rangle$ 가 k 번째 해를 나타내는 것을 의미한다. 만일 $\langle x:y \rangle$ 에 의해 표현되는 해가 없다면, 즉, $\langle x:y \rangle$ 가 유효하지 않은 표현이면, -1을 출력한다.

문제

철수의 토마토 농장에서는 토마토를 보관하는 큰 창고를 가지고 있다. 토마토는 아래의 그림과 같이 격자모양 상자의 칸에 하나씩 넣은 다음, 상자들을 수직으로 쌓아 올려서 창고에 보관한다.



탐색!

창고에 보관되는 토마토들 중에는 잘 익은 것도 있지만, 아직 익지 않은 토마토들도 있을 수 있다. 보관 후 하루가 지나면, 익은 토마토들의 인접한 곳에 있는 익지 않은 토마토들은 익은 토마토의 영향을 받아 익게 된다. 하나의 토마토에 인접한 곳은 위, 아래, 왼쪽, 오른쪽, 앞, 뒤 여섯 방향에 있는 토마토를 의미한다. 대각선 방향에 있는 토마토들에게는 영향을 주지 못하며, 토마토가 혼자 저절로 익는 경우는 없다고 가정한다. 철수는 창고에 보관된 토마토들이 며칠이 지나면 다 익게 되는지 그 최소 일수를 알고 싶어 한다.

토마토를 창고에 보관하는 격자모양의 상자들의 크기와 익은 토마토들과 익지 않은 토마토들의 정보가 주어졌을 때, 며칠이 지나면 토마토들이 모두 익는지, 그 최소 일수를 구하는 프로그램을 작성하라. 단, 상자의 일부 칸에는 토마토가 들어있지 않을 수도 있다.

입력

첫 줄에는 상자의 크기를 나타내는 두 정수 M, N 과 쌓아올려지는 상자의 수를 나타내는 H 가 주어진다. M 은 상자의 가로 칸의 수, N 은 상자의 세로 칸의 수를 나타낸다. 단, $2 \leq M \leq 100$, $2 \leq N \leq 100$, $1 \leq H \leq 100$ 이다. 둘째 줄부터는 가장 밑의 상자부터 가장 위의 상자까지에 저장된 토마토들의 정보가 주어진다. 즉, 둘째 줄부터 N 개의 줄에는 하나의 상자에 담긴 토마토의 정보가 주어진다. 각 줄에는 상자 가로줄에 들어있는 토마토들의 상태가 M 개의 정수로 주어진다. 정수 1은 익은 토마토, 정수 0은 익지 않은 토마토, 정수 -1은 토마토가 들어있지 않은 칸을 나타낸다. 이러한 N 개의 줄이 H 번 반복하여 주어진다.

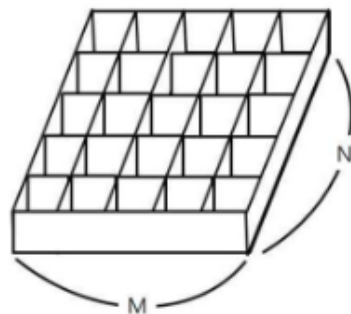
토마토가 하나 이상 있는 경우만 입력으로 주어진다.

출력

여러분은 토마토가 모두 익을 때까지 최소 며칠이 걸리는지를 계산해서 출력해야 한다. 만약, 저장될 때부터 모든 토마토가 익어있는 상태이면 0을 출력해야 하고, 토마토가 모두 익지는 못하는 상황이면 -1을 출력해야 한다.

11. 토마토

철수의 토마토 농장에서는 토마토를 보관하는 큰 창고를 가지고 있다. 토마토는 아래의 그림과 같이 격자 모양 상자의 칸에 하나씩 넣어서 창고에 보관한다.



탐색!

창고에 보관되는 토마토들 중에는 잘 익은 것도 있지만, 아직 익지 않은 토마토들도 있을 수 있다. 보관 후 하루가 지나면, 익은 토마토들의 인접한 곳에 있는 익지 않은 토마토들은 익은 토마토의 영향을 받아 익게 된다. 하나의 토마토의 인접한 곳은 왼쪽, 오른쪽, 앞, 뒤 네 방향에 있는 토마토를 의미한다. 대각선 방향에 있는 토마토들에게는 영향을 주지 못하며, 토마토가 혼자 저절로 익는 경우는 없다고 가정한다. 철수는 창고에 보관된 토마토들이 며칠이 지나면 다 익게 되는지, 그 최소 일수를 알고 싶어 한다.

토마토를 창고에 보관하는 격자모양의 상자들의 크기와 익은 토마토들과 익지 않은 토마토들의 정보가 주어졌을 때, 며칠이 지나면 토마토들이 모두 익는지, 그 최소 일수를 구하는 프로그램을 작성하라. 단, 상자의 일부 칸에는 토마토가 들어있지 않을 수도 있다.

입력

첫 줄에는 상자의 크기를 나타내는 두 정수 M, N 이 주어진다. M 은 상자의 가로 칸의 수, N 은 상자의 세로 칸의 수를 나타낸다. 단, $2 \leq M, N \leq 1,000$ 이다. 둘째 줄부터는 하나의 상자에 저장된 토마토들의 정보가 주어진다. 즉, 둘째 줄부터 N 개의 줄에는 상자에 담긴 토마토의 정보가 주어진다. 하나의 줄에는 상자 가로줄에 들어있는 토마토의 상태가 M 개의 정수로 주어진다. 정수 1은 익은 토마토, 정수 0은 익지 않은 토마토, 정수 -1은 토마토가 들어있지 않은 칸을 나타낸다.

토마토가 하나 이상 있는 경우만 입력으로 주어진다.

출력

여러분은 토마토가 모두 익을 때까지의 최소 날짜를 출력해야 한다. 만약, 저장될 때부터 모든 토마토가 익어있는 상태이면 0을 출력해야 하고, 토마토가 모두 익지는 못하는 상황이면 -1을 출력해야 한다.

12. 이중 우선순위 큐

이중 우선순위 큐

성공 다국어한국어 4 골드 IV

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
6 초	256 MB	37840	8700	6384	22.403%

문제

이중 우선순위 큐(dual priority queue)는 전형적인 우선순위 큐처럼 데이터를 삽입, 삭제할 수 있는 자료 구조이다. 전형적인 큐와의 차이점은 데이터를 삭제할 때 연산(operation) 명령에 따라 우선순위가 가장 높은 데이터 또는 가장 낮은 데이터 중 하나를 삭제하는 점이다. 이중 우선순위 큐를 위해선 두 가지 연산이 사용되는데, 하나는 데이터를 삽입하는 연산이고 다른 하나는 데이터를 삭제하는 연산이다. 데이터를 삭제하는 연산은 또 두 가지로 구분되는데 하나는 우선순위가 가장 높은 것을 삭제하기 위한 것이고 다른 하나는 우선순위가 가장 낮은 것을 삭제하기 위한 것이다.

정수만 저장하는 이중 우선순위 큐 Q가 있다고 가정하자. Q에 저장된 각 정수의 값 자체를 우선순위라고 간주하자.

Q에 적용될 일련의 연산이 주어질 때 이를 처리한 후 최종적으로 Q에 저장된 데이터 중 최댓값과 최솟값을 출력하는 프로그램을 작성하라.

입력

입력 데이터는 표준입력을 사용한다. 입력은 T개의 테스트 데이터로 구성된다. 입력의 첫 번째 줄에는 입력 데이터의 수를 나타내는 정수 T가 주어진다. 각 테스트 데이터의 첫 번째 줄에는 Q에 적용할 연산의 개수를 나타내는 정수 k ($k \leq 1,000,000$)가 주어진다. 이어지는 k 줄 각각엔 연산을 나타내는 문자('D' 또는 'I')와 정수 n이 주어진다. 'I n'은 정수 n을 Q에 삽입하는 연산을 의미한다. 동일한 정수가 삽입될 수 있음을 참고하기 바란다. 'D 1'은 Q에서 최댓값을 삭제하는 연산을 의미하며, 'D -1'은 Q에서 최솟값을 삭제하는 연산을 의미한다. 최댓값(최솟값)을 삭제하는 연산에서 최댓값(최솟값)이 둘 이상인 경우, 하나만 삭제됨을 유념하기 바란다.

만약 Q가 비어있는데 적용할 연산이 'D'라면 이 연산은 무시해도 좋다. Q에 저장될 모든 정수는 32-비트 정수이다.

출력

출력은 표준출력을 사용한다. 각 테스트 데이터에 대해, 모든 연산을 처리한 후 Q에 남아 있는 값 중 최댓값과 최솟값을 출력하라. 두 값은 한 줄에 출력하되 하나의 공백으로 구분하라. 만약 Q가 비어있다면 'EMPTY'를 출력하라.

힙큐!

문제

네 개의 명령어 D, S, L, R 을 이용하는 간단한 계산기가 있다. 이 계산기에는 레지스터가 하나 있는데, 이 레지스터에는 0 이상 10,000 미만의 십진수를 저장할 수 있다. 각 명령어는 이 레지스터에 저장된 n 을 다음과 같이 변환한다. n 의 네 자릿수를 d_1, d_2, d_3, d_4 라고 하자(즉 $n = ((d_1 \times 10 + d_2) \times 10 + d_3) \times 10 + d_4$ 라고 하자)

1. D: D 는 n 을 두 배로 바꾼다. 결과 값이 9999 보다 큰 경우에는 10000 으로 나눈 나머지를 취한다. 그 결과 값($2n \bmod 10000$)을 레지스터에 저장한다.
2. S: S 는 n 에서 1 을 뺀 결과 $n-1$ 을 레지스터에 저장한다. n 이 0 이라면 9999 가 대신 레지스터에 저장된다.
3. L: L 은 n 의 각 자릿수를 왼편으로 회전시켜 그 결과를 레지스터에 저장한다. 이 연산이 끝나면 레지스터에 저장된 네 자릿수는 왼편부터 d_2, d_3, d_4, d_1 이 된다.
4. R: R 은 n 의 각 자릿수를 오른편으로 회전시켜 그 결과를 레지스터에 저장한다. 이 연산이 끝나면 레지스터에 저장된 네 자릿수는 왼편부터 d_4, d_1, d_2, d_3 이 된다.

위에서 언급한 것처럼, L 과 R 명령어는 십진 자릿수를 가정하고 연산을 수행한다. 예를 들어서 $n = 1234$ 라면 여기에 L 을 적용하면 2341 이 되고 R 을 적용하면 4123 이 된다.

여러분이 작성할 프로그램은 주어진 서로 다른 두 정수 A와 B($A \neq B$)에 대하여 A를 B로 바꾸는 최소한의 명령어를 생성하는 프로그램이다. 예를 들어서 $A = 1234, B = 3412$ 라면 다음과 같이 두 개의 명령어를 적용하면 A를 B로 변환할 수 있다.

$1234 \rightarrow_L 2341 \rightarrow_L 3412$

$1234 \rightarrow_R 4123 \rightarrow_R 3412$

따라서 여러분의 프로그램은 이 경우에 LL 이나 RR 을 출력해야 한다.

n 의 자릿수로 0 이 포함된 경우에 주의해야 한다. 예를 들어서 1000 에 L 을 적용하면 0001 이 되므로 결과는 1 이 된다. 그러나 R 을 적용하면 0100 이 되므로 결과는 100 이 된다.

탐색! But 시간 초과 주의

입력

프로그램 입력은 T 개의 테스트 케이스로 구성된다. 테스트 케이스 개수 T 는 입력의 첫 줄에 주어진다. 각 테스트 케이스로는 두 개의 정수 A와 B($A \neq B$)가 공백으로 분리되어 차례로 주어지는데 A는 레지스터의 초기 값을 나타내고 B는 최종 값을 나타낸다. A 와 B는 모두 0 이상 10,000 미만이다.

출력

A에서 B로 변환하기 위해 필요한 최소한의 명령어 나열을 출력한다. 가능한 명령어 나열이 여러가지면, 아무거나 출력한다.

14. 1,2,3 더하기

1, 2, 3 더하기

성공다국어☆한국어 ▾3 실버 III

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초 (추가 시간 없음)	512 MB	88353	57852	39285	63.817%

문제

정수 4를 1, 2, 3의 합으로 나타내는 방법은 총 7가지가 있다. 합을 나타낼 때는 수를 1개 이상 사용해야 한다.

- 1+1+1+1
- 1+1+2
- 1+2+1
- 2+1+1
- 2+2
- 1+3
- 3+1

DP~!!~!

정수 n 이 주어졌을 때, n 을 1, 2, 3의 합으로 나타내는 방법의 수를 구하는 프로그램을 작성하시오.

입력

첫째 줄에 테스트 케이스의 개수 T 가 주어진다. 각 테스트 케이스는 한 줄로 이루어져 있고, 정수 n 이 주어진다. n 은 양수이며 11보다 작다.

출력

각 테스트 케이스마다, n 을 1, 2, 3의 합으로 나타내는 방법의 수를 출력한다.

패션왕 신해빈

성공

다국어

☆

한국어 ▾

3 실버 III

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	128 MB	23532	12851	11057	54.757%

문제

해빈이는 패션에 매우 민감해서 한번 입었던 옷들의 조합을 절대 다시 입지 않는다. 예를 들어 오늘 해빈이가 안경, 코트, 상의, 신발을 입었다면, 다음날은 바지를 추가로 입거나 안경대신 렌즈를 착용하거나 해야한다. 해빈이가 가진 의상들이 주어졌을때 과연 해빈이는 알몸이 아닌 상태로 며칠동안 밖에 돌아다닐 수 있을까?

입력

첫째 줄에 테스트 케이스가 주어진다. 테스트 케이스는 최대 100이다.

- 각 테스트 케이스의 첫째 줄에는 해빈이가 가진 의상의 수 $n(0 \leq n \leq 30)$ 이 주어진다.
- 다음 n 개에는 해빈이가 가진 의상의 이름과 의상의 종류가 공백으로 구분되어 주어진다. 같은 종류의 의상은 하나만 입을 수 있다.

모든 문자열은 1이상 20이하의 알파벳 소문자로 이루어져있으며 같은 이름을 가진 의상은 존재하지 않는다.

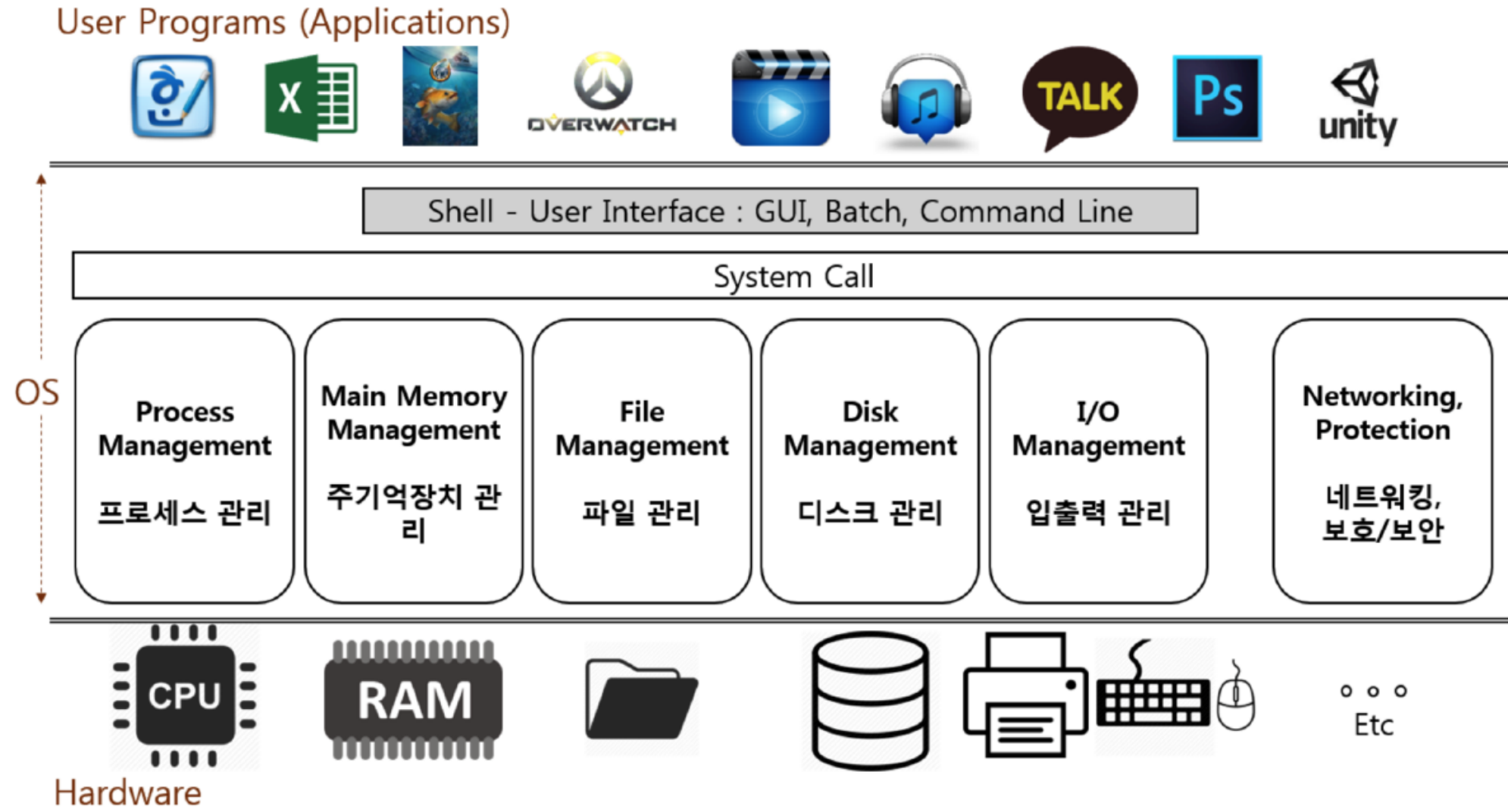
출력

각 테스트 케이스에 대해 해빈이가 알몸이 아닌 상태로 의상을 입을 수 있는 경우를 출력하시오.

한국인이 좋아하는 경우의 수!!



운영체제



<https://gyoogle.dev/blog/computer-science/operating-system/Process%20vs%20Thread.html>

<https://donghak-dev.tistory.com/101>

<https://www.infllearn.com/pages/weekly-infllearn-12>

<https://www.infllearn.com/pages/weekly-infllearn-12>

QnA