

# 대회 안내

## 1. 대회 관련

- 각 문제를 해결할 때마다 ({대회 후 경과시간}+{해당 문제 오답 횟수 x20분})의 패널티가 가산됩니다.
- 순위는 문제를 많이 푼 순 > 패널티가 적은 순 > 마지막 정답이 빠른 순으로 정해집니다.
- 문의하기기능을 사용해 문제와 직접적으로 관련된 사항에 대해 질문할 수 있습니다.

## 2. 제출관련

- 제출시에 문제 번호와 사용 언어를 정확히 선택해주세요. 참가자의 실수에 대해 책임지지 않습니다.
- 제출한 코드는 채점 현황에서 확인할 수 있습니다.

## 3. 채점 관련

- 표준 입출력 이외의 파일을 읽거나 쓰려고 시도하는 경우 런타임 에러로 처리하여 오답으로 인정합니다.
- 병렬연산이나 스레드를 이용하면 안됩니다. 모든 프로그램은 단일 스레드 프로그램이어야 합니다.
- 시스템에 영향을 주는 기능(종료, sleep 등)을 사용하면 무조건 오답처리 합니다.
- 채점 입력 데이터 모두에 대해 올바른 답을 출력하여야 합니다. 부분점수는 없습니다.
- 예제 데이터는 하나의 예제 일 뿐입니다. 실제 채점은 수 많은 비공개 데이터들로 이루어집니다.
- 각 채점 결과의 의미는 아래와 같습니다. 감독관은 오답의 이유를 절대 알려드리지 않습니다.

채점 결과 메시지	의미
ACCEPT	제출한 답안이 정답으로 인정된 경우
WRONG	제출한 답안이 오답으로 처리된 경우
TIMEOUT	제출한 프로그램이 시간 내에 처리되지 못 한 경우
COMPILE ERROR	제출한 코드에 문제가 있어서 컴파일 되지 않는 경우
RUNTIME ERROR	제출한 프로그램이 실행 도중에 문제를 일으킨 경우
PLEASE SELECT YOUR SUBMISSION LANGUAGE	<b>Language.txt에 채점 대상 언어가 명시되지 않은 경우</b> Language.txt에는 다음과 같은 키워드 중 하나를 제외하고는 어떠한 공백이나 줄넘김, 문자열들이 들어가는는 안됩니다 : c++, java, python (대소문자 구분)

## 4. C/C++ 관련

- 모든 입출력은 stdin, stdout을 통한 표준 입출력만 사용됩니다.
- GCC 컴파일러를 통해 채점됩니다. 비표준 MS C/C++ 라이브러리를 사용할 경우 컴파일 에러가 발생할 수 있습니다.
- C++의 cin/cout은 scanf/printf에 비하여 속도가 현저히 느립니다. 채점 시간에 반영되므로 후자의 이용을 권장합니다.

## 5. Java 관련

- 소스코드에 Package 선언문을 절대로 넣지 마세요. 오답처리됩니다.

## Hanyang ERICA Programming Contest 2017 - Maven Division

- 모든 입출력은 System.in, System.out을 통한 표준 입출력만 사용됩니다.
- Java의 특성상 문자열 처리 시 StringBuilder를 활용하는 편이 속도면에서 매우 유리합니다.
- System.in 입력 시 Scanner보다 BufferedReader와 InputStreamReader를 활용하는 편이 훨씬 속도가 빠릅니다.

C/C++ 레퍼런스 사이트 : <http://en.cppreference.com/w/>

Python 레퍼런스 사이트 : <http://python.org/>

JAVA 레퍼런스 사이트 : <http://docs.oracle.com/javase/7/docs/>

# Problem A

## 코딩변태

시간 제한 : 1 Second

눈을 떠보니 자신이 이상한 미로에 갇힌걸 발견한 성우는 눈 앞에 어떤 종이를 읽게 되었다.

난 전세계의 코딩변태들을 이곳에 모아서 그 중 누가 제일 변태인지를 가리는 테스트를 하기로 했다.  
규칙은 다음과 같다.

- 당신이 있는 시작점을 제외한 모든 방은 어둡다.
- 머물고 있던 방을 나서면 원래 있던 방은 절벽아래로 떨어진다.
- 어떤 방이든 들어선지 1분이 지나면 절벽아래로 떨어진다.
- 가장 오래 살아남은 사람이 이 세상의 최고의 코딩변태가 된다.

당신이 과연 최고의 코딩변태가 될 수 있을까?

자신의 목숨이 경각에 달했음을 깨닫지 못한 성우는 오히려 최고의 코딩변태가 되려고 한다. 성우의 전략은 다음과 같다.

“상하좌우의 우선순위를 뒤서 우선순위가 높은순서대로 무조건 이동하자!”

방향 우선순위와 미로의 상태가 주어졌을 때, 성우의 전략대로 진행할 경우 최대 몇분동안 살아남을 수 있는지를 구해주는 프로그램을 작성하시오.

### 입력

첫째 줄에 테스트 케이스의 개수  $T(1 \leq T \leq 16)$ 가 주어진다.

각각의 테스트 케이스의 첫 번째 줄에는 미로의 세로크기  $N$ 과 가로크기  $M$ 이 주어진다. 그 다음줄 부터  $N$ 개의 줄에는 미로의 상태가 주어진다. 미로의 상태는  $S$ 는 시작위치,  $.$ 는 낭떠러지,  $\#$ 은 방을 의미한다. 우선순위는 무조건 4글자이며  $W$ 는 위쪽,  $A$ 는 왼쪽,  $S$ 는 아래쪽,  $D$ 는 오른쪽을 의미한다.

$N, M$ 은 50보다 작거나 같은 자연수이다.

### 출력

각 테스트 케이스마다 살아남을 수 있는 최대 시간을 출력한다.

## 예제 입력

```
2
5 5
#####
#....
#.S##
#...#
#####
WASD
3 3
S##
.##
###
SAWD
```

## 예제 출력

```
17
5
```

## 힌트

성우는 최고의 코딩변태가 되지 못했다고 한다.

# Problem B

## 포켓몬 마스터 현주

시간 제한 : 20 Second

내 꿈은 포켓몬 마스터! 오늘도 포켓몬 마스터를 꿈꾸며 포켓몬 HO를 시작한 현주는 슬슬 그 꿈을 이루기 위해 주변에 있는 체육관을 살펴보기 시작한다. 그런데 마침 그 근처  $N \times N$ 개의 구역에 균일하게  $N \times N$ 개의 체육관이 있고 각 체육관의 대표 포켓몬의 전투력이 그 체육관의 좌표의 곱이라는 사실을 발견한다.

현주는 자신이 얼마나 포켓몬 마스터에 가까워졌나 확인하고 싶어서 각 체육관의 대표 포켓몬의 전투력을 강한 순서로 정렬한 후  $K$ 를 입력하면  $K$ 번째로 강력한 대표 포켓몬의 전투력을 알려주는 프로그램을 만드려고 한다.

포켓몬 마스터의 길은 멀고도 험하다는 사실을 알려주기 위해  $K$ 번째로 강력한 대표 포켓몬의 전투력을 알려주는 프로그램을 작성하시오.

### 입력

첫째 줄에 테스트 케이스의 개수  $T(1 \leq T \leq 1000)$ 가 주어진다.

각각의 테스트 케이스의 첫 번째 줄에는 구간의 길이  $N(0 < N \leq 100,000)$ 과  $K(0 < K \leq N \times N)$ 가 주어진다.

### 출력

각 테스트 케이스마다  $K$ 번째로 강력한 대표 포켓몬의 전투력을 출력한다.

#### 예제 입력

#### 예제 출력

2	6
3 2	6
5 15	

# Problem C

## 번복왕 주변복

시간 제한 : 1 Second

올해도 어김없이 1회, 2회를 거쳐 3회차 HEPC가 열리게 되었다. 올해는 무조건 결과를 반복하는 일이 없으리라 굳은 다짐을 하며, 이 문제 또한 출제를 하기 시작했다. 작년 재작년 번복사태를 겪어보지 않은 사람들을 위해 친절히 '굳이' 알려주려한다. 매 대회마다 대회 직후에 바로 결과를 알려주고, 실시간 순위를 확인하며 박진감 넘치는 대회 환경을 조성하려 했으나, 예상과 다른 일들이 하나 둘 터졌다. 첫 회에는 매우 급박하게(한달만에) 대회를 개최하다보니, 대회 채점 환경 테스트를 완벽하게 하지않아서(사실 굳이 확인 안해도 잘 되리라 믿었다) JAVA 재채점 사태가 일어났다. 대회 당일 JAVA 채점 결과가 친절히 나오지않아 JAVA 사용자들에게 많은 불편을 선사하고 말았다. 그리고 결과 발표 직후 JAVA 실수형 데이터 채점이 거지같이 되었다는 사실이 확인되어 모든 JAVA 제출 사항을 전면 재검토 재채점을 하고 진심어린 사과와 함께 결과를 재공지하였다. 2회차 HEPC에서는 절대로 반복하는 일이 없도록 해야겠다고 다짐을 하고, 작년에 비해 한달 늘어난 기간으로 대회를 준비하였다. 작년과 다르게 올해 문제는 자체 제작하였다. 꼼꼼히 검토하지않고 낸 문제 하나가 있었는데 예상 해답 알고리즘은 DP 였으나, 데이터 범위가 줄어드니 Brute Force로 풀리게 되었다. 다짐이 무색하게 또 한번의 좌절, 자괴감이 몰려왔다. 그리하여 3회차 HEPC는 내가 열지 말아야겠다고 생각을 했다. 하지만 대회 운영 가이드를 문서화하지 않아 이대로 손 놓으면 망할것이 뻔했다.(순전히 내 생각) 2017년의 3회차 HEPC는 1월부터 준비하여 체계적으로 완벽하게 준비했다. 그렇지만 사람 인생은 어찌될지 모르지 않는가. 또 한번의 번복사태가 일어나면, 일어나서는 안되지만, 이번 해는 진심으로 사과하는 의미로 시상식 날 사죄의 절을 하려한다. 이를 위해 바뀐 결과에 대해 절을 얼마나 해야할지 수치화하는 프로그램을 만들어 놓으려고 한다. 초기에 발표된 점수로 순위를 매겨 발표를 한 다음, 채점 오류가 발견되어 점수를 수정하여 다시 순위를 매긴다. 이때 초기 발표 순위와 번복 발표 순위를 비교해 얼마나 바뀌었는지에 판단하여 얼마나 절을 해야할지를 계산하는 프로그램이다. 문제의 배점은 1번이 1점, 2번이 10점, 3번이 100점, m번이  $10^{(m-1)}$ 점이다. 최종 점수 합이 같은 경우 해당하는 팀들은 같은 순위로 결정된다. 또한, 공동 순위 다음에 오는 순위는 공동 순위의 팀수가 더해진 등수로 결정된다. 만약 공동 2등이 10팀이라면, 다음 등수는 12등인 것이다. 계산 방법은 다음과 같다. 수상권이였다가 번복 이후 수상권에서 벗어난 경우 바뀐 등수의 제곱만큼 절을 하고, 수상권 내에서 낮은 수상권으로 이동한 경우 바뀐 등수의 두배만큼 절을 해야한다. 수상권 밖의 순위 변동은 고려하지 않는다. 예를 들면, 1등이였다가 10등으로 내려간 경우  $(10-1)^2$  이므로, 81회 절을 해야한다. 2등이였다가 3등으로 내려간 경우  $(3-2)^2$  이므로, 2회 절을 해야한다. 번복된 결과로 인해 총 얼마나 절을 해야하는지 계산해보자.

### 입력

첫 번째 줄에 개최된 대회의 횟수  $T(0 < T \leq 5)$ 가 주어진다. 각각의 테스트 케이스의 첫 줄에 각 대회의 참가자 수  $N$ , 문제의 개수  $M(0 < N \leq 1000, 0 < M \leq 30)$ 이 주어지고, 다음 줄부터 1번부터  $N$ 번 팀의  $M$ 개의 점수 정보가 주어진다. 이후 1번부터  $N$ 번 팀의  $M$ 개의 바뀐 점수에 대한 점수 정보가 주어진다. 점수는 100점 아니면 0점이기에 1 혹은 0으로 주어진다.

### 출력

각 대회에 절을 해야하는 횟수를 한 줄에 하나씩 출력한다.

## 예제 입력

```
2
11
1
0
10 4
1111
0111
1011
0011
1001
0001
1110
0110
1010
0010
0000
1011
1111
0111
1000
0100
1100
0010
1010
0110
```

## 예제 출력

```
0
83
```

\* 1,2회 HEPC 물의를 일으켜 진심으로 죄송하며, 순위 변동으로 인해 피해를 입은 분들께 정말로 죄송합니다.

# Problem D

## 쓰까문자

시간 제한 : 1 Second

은수는 HEPC 문제를 무엇을 낼지 고민을 하다가, 금강산도 식후경이라고, 점심으로 부산의 명물인 쓰까뽕밥을 먹으러 가기로 했다. 가게에 도착하니까 웬걸, 쓰까뽕밥집의 메뉴판은 음식들이 특정 규칙으로 쓰까써져 있었다. 문자를 쓰까쓰는 규칙은 다음과 같다. :

- 스까써진 문자열 X와 Z가 있다. X와 Z에 공통적으로 속하는 부분 문자열(subsequence) 중 가장 긴 문자열인 M을 찾으면, M이 은수가 먹고자하는 음식의 이름이다.
- 예를 들어, 문자열 '쓰까뽕밥'의 부분 문자열은 '쓰', '까', '뽕', '밥', '쓰까', '쓰뽕', '쓰밥', '까뽕', '까밥', '뽕밥', '쓰까뽕', '쓰까밥', '쓰뽕밥', '까뽕밥', '쓰까뽕밥' 이다.

은수를 도와 쓰까뽕밥집의 메뉴판을 해독해 은수가 먹을 수 있는 메뉴의 글자수를 알아내 보도록 하자.

### 입력

첫 번째 줄에 테스트 케이스 T( $1 \leq T \leq 10$ )가 주어진다. 각 테스트 케이스의 첫 번째 줄에는 쓰까뽕밥집의 메뉴 문자열 A, B가 공백으로 구분되어 주어진다. ( A,B의 길이  $\leq 10000$  ).

### 출력

각 테스트 케이스에 대해 은수가 먹을 수 있는 메뉴의 글자수를 출력한다.

#### 예제 입력

#### 예제 출력

2	0
AbCdE aBcDe	5
DunkinDunatsu TsukaDupbap	



# Problem E

## 성우의 강박증

시간 제한 : 1 Second

**강박 장애**(強迫障礙, Obsessive-Compulsive Disorder, **OCD**)는 자신의 의지와는 상관없이 어떤 특정한 사고나 행동을 떨쳐버리고 싶은데도 시도 때도 없이 반복적으로 하게 되는 상태를 말한다. 성우는 하얀 정사각형을 보면 밑변을 x축과 평행하게 두고 45도 회전시켜야 마음이 편해진다. 성우가 사랑하는 사각형의 모양은 다음과 같다.

<b>Size = 1</b>	<b>Size = 2</b>	<b>Size = 3</b>
0	0	0
	0 0	0 0
	0	0 0 0
		0 0
		0

어느 날 성우는 체스를 하다가 체스판의 흰색 타일을 이었을 때 자신을 만족시키는 사각형을 만들 수 있다는 것을 깨닫고 마음의 평안을 찾기 시작한다. 그걸 본 희재는 체스판을 내려치는데, 그 순간 놀랍게도 체스판의 검정색과 흰색 타일이 마구 섞이기 시작한다. 혼란스러워진 성우는 자신을 만족시키는 최대한 큰 사각형을 찾아야 함을 본능적으로 알아차렸다.

혼란스러워서 집중을 못하는 성우를 위해 성우가 원하는 가장 큰 사각형을 찾는 프로그램을 작성하시오.

### 입력

첫째 줄에 테스트 케이스의 개수  $T(1 \leq T \leq 18)$ 가 주어진다.

각각의 테스트 케이스의 첫 번째 줄에는 체스판의 크기  $N, M$ 이 주어진다.  $N, M$ 은 777보다 작거나 같은 자연수이다. 둘째 줄부터  $N$ 개의 줄에는 체스판의 상태가 주어진다.

### 출력

각 테스트 케이스마다 성우가 원하는 가장 큰 정사각형의 크기를 출력한다. 만약 하나도 찾지 못했다면 -1을 출력한다.

#### 예제 입력

```
2
5 5
10011
10100
00000
10000
00000
3 3
101
010
101
```

#### 예제 출력

```
3
2
```

# Problem F

## 개강총회

시간 제한 : 1 Second

설레는 마음으로 신입생을 만나러 개강총회에 참석한 현준은 빈 자리를 찾아 얼른 앉았다. 하지만 슬프게도 복학한 14학번 선배들이 가득한 자리였고, 하필 그 자리에 같은 대구 지역 출신인 준호가 앉아있었다. 준호는 파릇파릇한 신입생 시절부터 아재개그를 즐겨쳤다.

준호 : 세상에서 가장 지루한 중학교가 뭘줄 아냐?  
현준 : 아뇨; 모르겠는데요;  
준호 : 로딩중~ㅋㅋㅋㅋㅋㅋ  
현준 : ...

술을 먹고 더욱 업된 준호는 폭주하기 시작했다. 그리고 많이 취해버려 자신이 찼던 아재개그를 기억하지 못하고 반복해서 하기 시작했다. 옆자리에 앉아서 듣고 싶지 않아도 어쩔 수 없이 들어야 하는 상황인 현준은 해탈하고 이 상황을 즐길려고 노력했다. 준호가 반복해서 치는 아재개그에서 반복되는 부분의 최소 길이를 찾는 놀이를 만들어서 현준의 옆자리에 앉은 코딩번태 성우와 함께 즐겼다. 한편 준호가 술에 너무 취해버려 한글이 아닌 영어로 아재개그를 치기 시작했는데..

### 입력

첫 번째 줄에 테스트 케이스의 개수  $T(1 \leq T \leq 30)$ 가 주어진다. 각각의 테스트 케이스의 첫 줄에 아재개그의 길이  $(0 < s \leq 100,000)$ 가 주어진다. 두 번째 줄에 아재개그가 공백없이 영어 소문자로만 입력된다.

### 출력

첫 번째 줄에 아재개그의 반복되는 최소 길이를 출력한다.

#### 예제 입력

2	3
5	4
aabaa	
5	
aabcd	

#### 예제 출력

# Problem G

## 출구없는 매력

시간 제한 : 2 Second

939년 동안 사랑을 해본 적 없는 모태솔로 도깨비 김신은 꽃잎같이 하늘거리는 신부 지은탁에 의해 심장이 하늘에서 땅까지 아찔한 진자운동을 계속하게 되었다. 그 결과 939년만에 부정맥에 걸린 도깨비 김신은 엠블런스에 실려 병원으로 이송되었다. 펍 난감하게도 도깨비의 대저택은 사람들의 출입이 흔하지 않은 깊은 산골짜기에 있어서 병원으로 이송하는 길을 마치 미로와도 같았다. 이송되는 과정에서 산에 가로막혀 더 이상 갈 수 없는 상황이 발생하였다. 이때 김신은 마지막 힘을 짜내어 눈 앞에 존재하는 산을 없애고 정신을 잃고 말았다. 김신에게 이제 남은 생이 몇분 남지 않았다. 골든 아워 내에 김신이 병원으로 무사히 갈 수 있는지 알아보자.

### 입력

첫 번째 줄에 테스트 케이스의 수  $K(0 < K \leq 10)$ 가 주어진다. 각각의 테스트 케이스의 첫 줄에 미로와 같은 산골짜기의 가로  $N$ , 세로  $M(0 < N, M \leq 100)$ 의 크기와 김신이 버틸 수 있는 시간  $T(0 \leq T \leq N \cdot M)$ 초가 주어진다. 다음  $N$ 줄에  $M$ 개의 각 지점의 정보가 주어진다. 0은 길이며, 1은 산이다. 김신은 남아있는 마지막 힘을 짜내 단 한번 자신의 앞에 있는 산을 없애버릴 수 있다. 산골짜기의 시작은 왼쪽 위이며, 끝은 오른쪽 아래이다. 길을 한 블록 이동하는 시간은 1초이다. 시작지점에서 시간은 1초이다. 시작 지점과 끝 지점의 정보는 항상 0이다.

### 출력

$T$ 시간 내에 김신이 병원에 도착할 수 있다면 "Because it's a nice day", 만약 도착할 수 없다면 "Because it's a bad day"를 출력한다.

### 예제 입력

### 예제 출력

1	Because it's a bad day
5 5 10	
01111	
01000	
00010	
11111	
11110	

# Problem H

## 열쇠 찾기

시간 제한 : 1 Second

호연이는 제 3공학관의 출입을 담당하는 조교로써 임명되었다. 오늘은 위대한 컴퓨터 과학자인 에클레르 케익스트라가 3공학관 317호에서 세미나를 하기로 되어 있기 때문에, 문을 미리 열어 두어야 한다. 하지만 안타깝게도 전 출입 담당 조교인 제형이는 굉장히 보안을 신경쓰는 사람이라서, 문을 열기 위해서는 4개의 카드키를 찾아야만 문을 열 수 있게 해줬다. 제형이가 남긴 메모에 따르면, 각 카드키는 64비트의 데이터를 가지고 있는데 4개의 카드키의 데이터를 XOR해서 0이 되는 카드가 3공학관의 마스터키라고 한다. 호연이를 도와서 3공학관의 마스터키를 찾는 프로그램을 작성하도록 하자.

### 입력

첫 번째 줄에 테스트 케이스  $T(1 \leq T \leq 300)$ 가 주어진다. 각 테스트 케이스의 첫 번째 줄에는 카드의 갯수  $N$ 이 주어진다. ( $4 \leq N \leq 100000$ ). 다음 줄에는  $N$ 개의 양의 정수가 공백으로 분할되어 주어진다.

### 출력

각 테스트 케이스에 대해 호연이가 문을 열 수 있으면 DIJKSTRA를, 문을 열 수 없으면 THE CAKE IS LIE를 출력한다.

#### 예제 입력

#### 예제 출력

1 5 1 0 2 3 7	DIJKSTRA
---------------------	----------

# Problem I

## 내부자

시간 제한 : 2 Second

자신을 제 3공학관의 출입 관리 조교에서 해고한 이규정 교수에게 앙심을 품은 제형이는 이규정 교수의 방에 몰래 들어가 교수의 연구자료를 모두 훔칠 사악한 계획을 세운다. 하지만 이규정 교수의 연구실은 4공학관에 있어서 아무에게도 들키지 않게 들어가기 위해서는 잠금 장치를 모두 해킹해서 무력화해야 한다. 제형이는 ‘버퍼 오버플로우’라는 기술을 이용해서 잠금장치를 해킹하려고 하는데 그 방식은 다음과 같다.

- 잠금장치는 K개의 카드 슬롯을 가지고 있다.
- N개의 카드가 있을 때, 이 카드들을 K개의 카드 슬롯에 나누어 꽂게 된다. 단, 여러 카드를 한 슬롯에 꽂게 될 경우, 카드의 데이터는 카드 데이터들을 모두 OR한 값과 같다.
- K의 카드 슬롯에 입력된 카드 데이터들의 합이 잠금 장치의 버퍼 B보다 크면 잠금장치는 고장을 일으키며 잠금이 해제된다.

제형이를 도와 이규정 교수의 연구실에 침입이 가능한지 알아내는 프로그램을 작성하자.

### 입력

첫 번째 줄에 테스트 케이스  $T(1 \leq T \leq 300)$ 가 주어진다. 각 테스트 케이스의 첫 번째 줄에는 제형이가 가진 위조 카드의 갯수  $N(1 \leq N \leq 1000)$ 과 이규정 교수 연구실 잠금장치의 카드 슬롯 갯수  $K(1 \leq K \leq N)$ , 잠금 장치의 버퍼 크기  $B(0 \leq B \leq 2^{30})$ 가 주어진다. 다음 줄에는 N개의 카드 데이터가 공백으로 구분되어 주어진다.

### 출력

각 테스트 케이스에 대해 제형이가 이규정 교수 연구실에 침입 가능하면 BURN EVERYTHING을, 침입 불가능하면 WASTED를 출력한다

#### 예제 입력

```
4
3 2 6
1 2 2
4 3 8
1 2 3 4
2 2 10
1 2
11 4 704
66 152 7 89 42 28 222 69 10 54 99
```

#### 예제 출력

```
WASTED
BURN EVERYTHING
WASTED
WASTED
```

# Problem J

## 문제 출제하기

시간 제한 : 1 Second

야 내가 그 엘리스 문제 출제 회의하다가 개쩌는 문제 발견했는데 함 들어봐봐.  
선생님이 이제  $m$ 번 숫자를 부를거야. 그럼 불린 학생은 맨 뒤로 가야됨.  $n$ 이 5라고하면 1 2 3 4 5 있을때, 3부르면 1 2 4 5 3 되는거지. 그럼 이제  $m$ 번 숫자를 다 부르고나면 학생들이 어떻게 배치가 되겠지. 근데 이제 여기서 선생이 뭐하고싶냐면 어느 학생이 가장 멀리 떨어져 봤는지를 알고 싶은거임. 예를 들어서, 1 2 3 4 5 에서 2를 부르면 1 3 4 5 2 가 되니까 3, 4, 5는 제자리에서 1만큼 떨어진거고 2는 제자리에서 3만큼 떨어진거지. 여기서 또 4를 부르면 1 3 5 2 4 가 되고 3은 제자리에서 1만큼, 5는 제자리에서 2만큼, 2는 제자리에서 2만큼, 4는 제자리에서 1만큼 떨어진게 되는거지. 지금까지 보면 2는 3만큼 떨어져봤고, 3은 1만큼만 떨어져봤고, 4는 1만큼 떨어져봤고, 5는 2만큼 떨어져본거지. 그럼  $m$ 번 숫자 부를때 누가 제일 처음 위치에서 멀리 떨어져봤냐?는거 어떤데 한번 짜봐라.

### 입력

입력 데이터는 표준입력을 사용한다. 입력은  $T(0 < T \leq 10)$ 개의 테스트 데이터로 구성된다. 입력의 첫 번째 줄에는 테스트 데이터의 개수  $T$ 가 정수로 주어진다. 각 테스트 데이터의 첫 번째 줄에 학생의 명수  $n(1 \leq n \leq 1000)$ 과 선생님이 학생을 부른 횟수  $m(1 \leq m \leq 1000)$ 이 주어진다. 두 번째 줄에는  $m$ 개의 불린 학생의 번호가 공백을 통해 구분되어 주어진다. 학생은 번호 순서대로 서있다고 가정한다.

### 출력

출력은 표준출력을 사용한다. 각 테스트 데이터에 대해, 첫 번째 줄에 가장 멀리 떨어져 본 학생의 번호를 출력한다. 만약 멀리 떨어져 본 거리가 같은 학생이 여러명이라면 학생 번호가 가장 작은 숫자를 출력한다.

#### 예제 입력

1 5 2 2 4	2
-----------------	---

#### 예제 출력

# Problem K

## 이솜우화

시간 제한 : 1 Second

날이 좋은 어느날 달리기 연습을 하는 거북이를 보고, 토끼가 크게 웃었습니다. 비웃는 소리를 듣고 화가난 거북이는 토끼에게 달리기 경주를 신청하였습니다. "준비, 땅 !" 을 하게 되면 둘은 시작 지점에서 동시에 출발하게 됩니다. 당근을 매우 좋아하는 토끼는 경기 당일 당근 밭의 유혹을 견딜 수 없을 것입니다. 당근밭을 밟게 되면 당근을 먹느라 보통이면 1분이면 지날 거리를 2분동안 지나게 됩니다. 그리고 오만한 토끼는 도착 지점까지 지나가는 길에 존재하는 모든 당근을 먹고 지나가도 거북이를 이길것이라 생각할 것입니다. 한편, 거북이는 열심히 노력하지만 길을 지날 때 평소의 토끼보다 2배의 시간이 걸립니다. 그리고 껍질외의 피부 부분이 약하기 때문에 당근밭을 지나갈 수 없습니다. 승부에 눈이 먼 거북이는 경기 시작 전 경기의 지도를 미리 입수해 놓았습니다. 영악한 거북이는 경기 시작 전 경기 결과를 미리 알고 대회 장소를 옮기고자 합니다. 대회 장소가 주어졌을 때, 거북이가 이길 수 있는지 없는지 출력하는 프로그램을 작성하세요.

### 입력

첫 번째 줄에 테스트 케이스의 수  $T(1 \leq T \leq 12)$ 가 주어진다. 각각의 테스트 케이스의 첫 줄에 맵의 크기  $N(1 < N < 100)$ 이 주어진다. 다음  $N$ 개의 줄에  $N$ 개의 지도의 정보가 주어진다. 정보는 출발지  $S$ , 도착지  $D$ , 길  $L$ , 당근밭  $C$ 으로 표현된다. 한 블록을 이동하는데 1분의 시간이 소요되고, 토끼만 대각선으로도 이동할 수 있다. (단, 시작점에서의 시간은 1(분)이다.)

### 출력

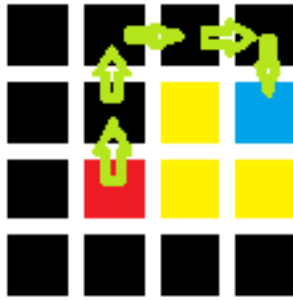
한 줄에 거북이가 이기는 경기면 'WIN', 거북이가 지는 경기면 'LOSE', 무승부인 경우는 'DRAW'를 출력한다.

### 예제 입력

### 예제 출력

1 4 LLLL LLCD LSCC LLLL	LOSE
--	------

## 힌트



[그림1] 거북이의 경로



[그림2] 토끼의 경로



# Problem L

## 수심

시간 제한 : 1 Second

나는 촌장 부인이다. 우리 마을은 다른 마을과 수심(殊甚)<sup>1</sup>하게 어업으로 생을 이어간다. 물고기를 잡기 위해 바다 위에는 3개의 부표가 띄어져 있고, 부표는 각각 줄로 이어져 있다. 수심가지인심난지(水深可知人心難知)<sup>6</sup>라는 말이 있듯, 물고기의 마음은 알 수가 없다. 그러나 물고기들은 수심(水深)<sup>2</sup>이 깊으면 깊을 수록 더 많이 잡힌다고 한다. 우리 마을 주변 바다 중에서 가장 수심(水深)<sup>2</sup>이 깊은 곳은 단순히 수심(水心)<sup>4</sup>이 아닌 세 개의 부표를 꼭지점으로 하는 삼각형의 수심(垂心)<sup>3</sup>이다. 지구온난화로 인해 어업으로 얻는 수익이 악화되어 장탄수심(長歎愁心)<sup>5</sup>을 하는 우리 촌장을 위해 우리가 물고기가 많이 있는 곳을 알려주자.

### 입력

첫째 줄에 A 부표를 제외한 B 부표의 좌표  $x_1(0 \leq x_1 < 10000)$ ,  $y_1(0 \leq y_1 < 10000)$ 가 공백을 통해 구분되어 입력된다. 두번째 줄에 C 부표의 좌표  $x_2(0 \leq x_2 < 10000)$ ,  $y_2(0 \leq y_2 < 10000)$ 가 공백을 통해 구분되어 입력된다. (A 부표의 위치는 원점(0, 0)이다. 또한, 부표가 서로 겹치는 경우는 없다고 한다.)

### 출력

물고기가 많이 잡히는 곳(垂心)의 좌표  $x, y$ 를 줄바꿈을 통해 구분하여 출력한다. 만약 수심을 구할 수 없는 경우 -1을 출력한다. (소수점은 여섯번째 자리에서 반올림 한다.)

### 예제 입력

### 예제 출력

3 0	2.00000
2 7	0.28571

### 용어

殊甚<sup>1</sup> : 유별나게 다름

水深<sup>2</sup> : 물의 깊이

垂心<sup>3</sup> : 삼각형(三角形)의 각 꼭짓점으로부터 그 맞은편 변에 그은 세 수선(垂線)이 서로 만나는 점(點)

水心<sup>4</sup> : 수면의 중심

長歎愁心<sup>5</sup> : 크게 탄식(歎息)하며 하는 근심

水深可知人心難知<sup>6</sup> : 물의 깊이는 알 수 있으나, 사람의 속마음은 헤아리기 어렵다는 말