

21981번 - C번 - 고장난 시계

26점 서버태스크

한국어 ▾

시간 제한	메모리 제한
3 초 (하단 참고)	512 MB

문제

Albert는  $n$ 개의 전자 탁상 시계를 가지고 있다 (편의상 1, 2, ...,  $n$ 으로 번호가 붙어있다).

각 시계는 현재 시각을 "HH:MM:SS"의 방식으로 시:분:초로 보여주는데 항상  $0 \leq HH < 24, 0 \leq MM < 60, 0 \leq SS < 60$ 을 만족한다. 만약 HH, MM, SS 값이 10 미만인 경우 선행0을 표시하여 각 시계에는 언제나 6개의 숫자가 표시된다. 예를 들어, "00:00:01" 은 자정에서 1초가 지난 시각이고, "12:00:00"는 정오를 나타내며, "18:05:05"는 저녁 6시에서 5분 5초가 지난 시각을 나타낸다.

00:00:0112:00:0018:05:05

각 시계는 현재 가리키는 시각도 제각각이고, 고장난 것도 있어서 매초 1초가 아닌 다른 시간만큼 진행되기도 한다. 구체적으로,  $i$ 번째 시계가 현재 가리키고 있는 시각을  $T[i]$ 라 하고 매초 이 시계는  $D[i]$ 초 이후의 시각을 보여 준다 하자 (즉, 1초마다  $D[i]$ 초씩 증가한다).

예를 들어,  $n = 3$ 이고  $T = [ "11:12:00", "11:12:20", "11:12:40" ]$  그리고  $D = [4, 2, 0]$ 이라 하자.

- 현재 각 시계는 다른 시각을 보여주고 있다.
- 현재로부터 5초가 지난 후, 1번 시계는 "11:12:20", 2번 시계는 "11:12:30", 3번 시계는 "11:12:40"을 보여 준다.
- 현재로부터 10초가 지난 후, 세 시계는 모두 "11:12:40"을 보여 준다 (이 때  $n$ 개의 시계가 모두 "동기화" 되었다고 한다).
- 현재로부터 43210초가 지난 후, 세 시계는 한 번 더 동기화 된다.

이 예제의 경우, 24시간 동안 세 시계는 정확히 두 번 동기화 된다.

Albert는  $n$ 개의 탁상 시계가 현재 가리키는 시각과 각 시계가 매초 몇초씩 진행하는지 정보를 이용하여, 앞으로 24시간 (=86400초) 동안  $n$ 개의 시계가 정확히 몇 번 동기화 될지 계산해보고 싶다. (힌트 참고)

입력

첫 줄에 테스트 케이스의 수  $T$ 가 주어진다.

각 테스트 케이스는 세 줄에 나누어 주어진다.

테스트 케이스의 첫 줄에는  $n$ 이 주어진다.

둘째 줄에는  $n$ 개의 시계가 현재 보여주는 시각이 ( $T[i]$ ) 공백으로 구분되어 "HH:MM:SS" 형식의 문자열로 주어진다.

셋째 줄에는  $n$ 개의 시계가 매초 몇 초씩 진행하는지 ( $D[i]$ ) 공백으로 구분되어 주어진다.

출력

각 테스트 케이스의 정답을 각 줄에 출력한다.

제한

- $1 \leq T \leq 20$
- $2 \leq n \leq 70,000$
- $T[i]$  는 언제나 "HH:MM:SS"의 형식으로 주어지며  $0 \leq HH < 24, 0 \leq MM < 60, 0 \leq SS < 60$  을 만족한다 (HH, MM, SS는 모두 정수이며 10 미만의 경우 선행 0이 하나 붙어 주어진다).
- $-10^9 \leq D[i] \leq 10^9$

서브태스크 1 (7점)

- $2 \leq n \leq 170$

## 서브태스크 2 (19점)

- $2 \leq n \leq 70,000$

### 예제 입력 1 복사

```
6
3
11:12:00 11:12:20 11:12:40
4 2 0
2
00:00:00 23:59:59
1 1
2
00:00:00 00:00:00
90000 3600
2
11:00:00 11:00:00
1 -1
3
11:00:00 11:00:00 23:00:00
1 -1 0
3
15:19:59 16:07:49 15:44:54
966 392 667
```

### 예제 출력 1 복사

```
2
0
86400
2
1
1
```

예제 1: 본문에서 다루었다.

예제 2: 두 시계는 각자 매초 1초씩 진행하므로 24시간 동안 한 번도 동기화 되지 않는다.

예제 3: 이 두 시계는 언제나 같은 시각을 보여주므로 24시간 동안 총 86400번 동기화 된다 (24시간 = 86400초).

예제 4: 이 두 시계는 12시간마다 한 번 동기화 된다.

예제 5: 추가 설명 없음.

예제 6: 추가 설명 없음.

## 힌트

이 문제는 임의의 24시간 (= 86400초) 동안  $n$ 개의 시계가 동기화 되는 회수를 세면 된다. 현재 시각을 "0초 후"라 보면 0초 후 부터 86399초 후 까지  $n$ 개의 시계가 동기화 되는 회수를 세어도 되고 (예제 3, 4 참고) 혹은 1초 후 부터 86400초 후 까지  $n$ 개의 시계가 동기화 되는 회수를 세어도 된다. 마찬가지로  $s$ 초 후 부터  $(s+86399)$ 초 후 까지  $n$ 개의 시계가 동기화 되는 회수를 세어도 된다. 어떤 방법을 택하더라도 정답은 같다.

## 시간 제한

- Java 8: 4 초
- PyPy3: 5 초