

## TEST 12

1.

주차장의 요금표와 차량이 들어오고(입차) 나간(출차) 기록이 주어졌을 때, 차량별로 주차 요금을 계산하려고 합니다. 아래는 하나의 예시를 나타냅니다.

# 요금표

기본 시간(분)	기본 요금(원)	단위 시간(분)	단위 요금(원)
180	5000	10	600

# 입/출차 기록

시각( 시 : 분 )	차량 번호	내역
05 : 34	5961	입차
06 : 00	0000	입차
06 : 34	0000	출차
07 : 59	5961	출차
07 : 59	0148	입차
18 : 59	0000	입차
19 : 09	0148	출차
22 : 59	5961	입차
23 : 00	5961	출차

# 자동차별 주차 요금

차량 번호	누적 주차 시간(분)	주차 요금(원)
0000	$34 + 300 = 334$	$5000 + [(334 - 180) / 10] \times 600 = 14600$
0148	670	$5000 + [(670 - 180) / 10] \times 600 = 34400$
5961	$145 + 1 = 146$	5000

## TEST 12

# 어떤 차량이 입차된 후에 출차된 내역이 없다면, 23:59에 출차된 것으로 간주합니다.

> 0000 번 차량은 18:59에 입차된 이후, 출차된 내역이 없습니다. 따라서, 23:59에 출차된 것으로 간주합니다.

# 00:00 부터 23:49 까지의 입/출차 내역을 바탕으로 차량별 누적 주차 시간을 계산하여 요금을 일괄로 정산합니다.

# 누적 주차 시간이 기본 시간 이하라면, 기본 요금을 청구합니다.

# 누적 주차 시간이 기본 시간을 초과하면, 기본 요금에 더해서, 초과한 시간에 대해서 단위 시간마다 단위 요금을 청구합니다.

> 초과한 시간이 단위 시간으로 나누어 떨어지지 않으면, 올림 합니다.

> [a]: a보다 작지 않은 최소의 정수를 의미합니다. 즉, 올림 을 의미합니다.

주차 요금을 나타내는 정수 배열 fees, 자동차의 입/출차 내역을 나타내는 문자열 배열 records 가 매개변수로 주어집니다. 차량 번호가 작은 자동차부터 청구할 주차 요금을 차례대로 정수 배열에 담아서 return 하도록 solution 함수를 완성해주세요.

## TEST 12

### 제한사항

▶ `fees`의 길이 = 4

> `fees[0]` = 기본 시간(분) /  $1 \leq \text{fees}[0] \leq 1,439$

> `fees[1]` = 기본 요금(원) /  $0 \leq \text{fees}[1] \leq 100,000$

> `fees[2]` = 단위 시간(분) /  $1 \leq \text{fees}[2] \leq 1,439$

> `fees[3]` = 단위 요금(원) /  $1 \leq \text{fees}[3] \leq 10,000$

▶  $1 \leq \text{records}$ 의 길이  $\leq 1,000$

> `records`의 각 원소는 “시각 차량번호 내역” 형식의 문자열입니다.

> 시각, 차량번호, 내역은 하나의 공백으로 구분되어 있습니다.

> 시각은 차량이 입차되거나 출차된 시각을 나타내며, HH:MM 형식의 길이 5인 문자열입니다.

- HH:MM 은 00:00 부터 23:59 까지 주어집니다.

> 차량번호는 자동차를 구분하기 위한, 0~9로 구성된 길이 4인 문자열입니다.

> 내역의 IN 은 입차를, OUT은 출차를 의미합니다.

> `records`의 원소들은 시각을 기준으로 오름차순으로 정렬되어 주어집니다.

> `records`는 하루 동안의 입/출차된 기록만 담고 있으며, 입차된 차량이 다음날 출차되는 경우는 입력으로 주어지지 않습니다.

> 같은 시각에, 같은 차량번호의 내역이 2번 이상 나타내지 않습니다.

> 마지막 시각(23:59)에 입차되는 경우는 입력으로 주어지지 않습니다.

> 아래의 예를 포함하여, 잘못된 입력은 주어지지 않습니다.

- 주차장에 없는 차량이 출차되는 경우

- 주차장에 이미 있는 차량 (차량번호가 같은 차량) 이 다시 입차되는 경우

## TEST 12

fees	records	result
180 5000 10 600	["05:34 5961 IN", "06:00 0000 IN", "06:34 0000 OUT", "07:59 5961 OUT", "07:59 0148 IN", "18:59 0000 IN", "19:09 0148 OUT", "22:59 5961 IN", "23:00 5961 OUT"]	14600, 34400, 5000
120 0 60 591	["16:00 3961 IN", "16:00 0202 IN", "18:00 3961 OUT", "18:00 0202 OUT", "23:58 3961 IN"]	0, 591
1 461 1 10	["00:00 1234 IN"]	14841

#1> 문제 예시와 같습니다.

#2>

차량 번호	누적 주차 시간(분)	주차 요금(원)
0202	120	0
3961	120 + 1	$0 + [(121 - 120) / 60] \times 591 = 591$

: 3961번 차량은 2번째 입차된 후, 출차된 내역이 없으므로, 23:59에 출차되었다고 간주합니다.

#3>

차량 번호	누적 주차 시간(분)	주차 요금(원)
1234	1439	$461 + [(1439 - 1) / 1] \times 10 = 14841$

: 1234번 차량은 출차 내역이 없으므로, 23:59에 출차되었다고 간주합니다.

## TEST 12

2. 진하의 팀에서는 기능 개선 작업을 수행 중입니다. 각 기능은 진도가 100% 일 때 서비스에 반영할 수 있습니다. 또, 각 기능의 개발속도는 모두 다르기 때문에 뒤에 있는 기능이 앞에 있는 기능보다 먼저 개발될 수 있고, 이때 뒤에 있는 기능은 앞에 있는 기능이 배포될 때 함께 배포됩니다.

먼저 배포되어야 하는 순서대로 작업의 진도가 적힌 정수 배열 progresses와 각 작업의 개발 속도가 적힌 정수 배열 speeds가 주어질 때 각 배포마다 몇 개의 기능이 배포되는지를 return 하도록 solution 함수를 완성해주세요.

### 제한사항

- ▶ 작업의 개수는 100개 이하입니다.
- ▶ 작업의 진도는 100 미만의 자연수입니다.
- ▶ 작업의 속도는 100 이하의 자연수 입니다.
- ▶ 배포는 하루에 한 번만 할 수 있으며, 하루의 끝에 이루어진다고 가정합니다. 예를 들어 진도율이 95%인 작업의 개발 속도가 하루에 4%라면 배포는 2일 뒤에 이루어집니다.

### 입출력 예

progresses	speeds	return
93 30 55	1 30 5	2 1
95 90 99 99 80 99	1 1 1 1 1 1	1 3 2

### 입출력 예 설명

#2> 모든 기능이 하루에 1%씩 작업이 가능하므로, 작업이 끝나기까지 남은 일수는 각각 5일, 10일, 1일, 1일, 20일, 1일 입니다. 어떤 기능이 먼저 완성되었더라도 앞에 있는 모든 기능이 완성되지 않으면 배포가 불가능합니다.

따라서 5일째에 1개의 기능, 10일째에 3개의 기능, 20일째에 2개의 기능이 배포됩니다.