

문제 설명

가위바위보 대회가 열렸습니다. n 명의 사람이 참가한 이 대회는 m 개의 라운드로 구성되어 있으며, 매 라운드마다 획득할 수 있는 점수가 있습니다. 대회 결과, 최고득점자가 얻은 점수를 구하려고 합니다.

매 라운드마다 모든 참가자는 가위(S), 바위(R), 보(P)중에서 한 가지 모양을 선택해야 합니다. 그리고 같은 모양을 낸 사람들을 한 그룹으로 묶습니다. 어떤 그룹이 라운드에 걸린 점수를 획득했다면, 그 그룹에 속한 모든 참가자가 해당 점수를 얻습니다. 그룹의 크기는 그룹에 속한 참가자의 수와 같습니다.

라운드에 걸리는 점수는 정수이며, 음수가 될 수도 있습니다. 참가자들이 낸 모양의 종류 수와, 라운드 점수에 따라 어떤 그룹이 점수를 가져갈지 결정됩니다.

- 모양이 한 가지인 경우: 바로 다음 라운드로 넘어갑니다. 이번 라운드에 걸렸던 점수는 다음 라운드의 점수와 합칩니다.
- 모양이 두 가지인 경우
 - 라운드에 걸린 점수가 0 이상인 경우: 두 그룹 중에서 가위바위보를 이긴 그룹이 점수를 가져갑니다.
 - 라운드에 걸린 점수가 음수인 경우: 두 그룹 중에서 가위바위보를 진 그룹이 점수를 가져갑니다.
- 모양이 세 가지인 경우
 - 세 그룹의 크기가 모두 같은 경우: 바로 다음 라운드로 넘어갑니다. 이번 라운드에 걸렸던 점수는 다음 라운드의 점수와 합칩니다.
 - 두 그룹의 크기가 같은 경우: 크기가 같은 두 그룹을 제외한 나머지 한 그룹이 점수를 가져갑니다.
 - 그룹의 크기가 모두 다른 경우
 - 라운드에 걸린 점수가 0 이상인 경우: 그룹의 크기가 가장 큰 그룹을 제외한 나머지 두 그룹 중에서 가위바위보를 이긴 그룹이 점수를 가져갑니다.
 - 라운드에 걸린 점수가 음수인 경우: 그룹의 크기가 가장 작은 그룹을 제외한 나머지 두 그룹 중에서 가위바위보를 진 그룹이 점수를 가져갑니다.

마지막 라운드를 넘길 경우, 마지막 라운드에 걸린 점수는 어느 그룹도 가져갈 수 없습니다.

참가자 수를 나타내는 정수 `n`, 라운드 수를 나타내는 정수 `m`, 라운드에 걸린 점수를 순서대로 담은 정수 배열 `points`, 각 라운드에 참가자들이 낼 모양을 담은 문자열 배열 `hands`가 매개변수로 주어집니다. 대회 결과, 최고 득점자가 얻은 점수를 return 하도록 solution 함수를 완성해주세요.

제한사항

- $2 \leq n \leq 1,000$
- $1 \leq m \leq 1,000$
- `points`의 길이 = `m`
 - $-100 \leq \text{points의 원소} \leq 100$
- `hands`의 길이 = `m`
 - `hands`의 원소는 길이가 `n`인 문자열입니다.
 - 문자열은 'S', 'R', 'P'로 이루어져 있습니다.
 - 'S'는 가위를, 'R'은 바위를, 'P'는 보를 나타냅니다.
 - `hands[i][j]`는 `i+1`번째 라운드에 `j+1`번째 참가자가 선택한 모양을 나타냅니다.

입출력 예

n	m	points	hands	result
6	5	[5, -2, 1, 3, -5]	["PSPRSS", "SSRRSS", "RRRRRR", "RRSSPP", "SSSRRP"]	4

입출력 예 설명

입출력 예 #1

두 번째 게임의 초기 배열은 $[0, 1, 0, 1]$ 입니다.

위와 같은 이유로 이번 게임에서는 첫 번째 플레이어가 무조건 패배합니다. 따라서 0 을 정답 배열에 추가합니다.

따라서 정답으로 $[1, 0]$ 을 return 해야 합니다.

문제 설명

팰린드롬(Palindrome)이란 앞에서부터 읽거나 뒤에서부터 읽었을 때 둘 다 같은 결과가 나오는 배열을 말합니다.

`[0, 0, 1, 0, 0]` , `[1, 0, 0, 1]` , `[3, 7, 3]` 은 팰린드롬이지만, `[1, 2, 3]` , `[0, 0, 1, 0]` , `[1, 3, 2]` 는 팰린드롬이 아닙니다.

팰린드롬 게임을 하려고 합니다. 다음은 팰린드롬 게임의 설명입니다.

1. 게임은 턴제로 진행되며, 항상 두 명이 플레이합니다.
2. 초기에 0 이상 9 이하의 정수가 들어있는 배열이 하나 주어집니다. 양 플레이어는 하나의 배열을 공유하며 플레이합니다.
3. 각 플레이어는 자신의 턴이 되면 다음과 같은 행동을 해야 합니다.
 1. 배열에서 **0이 아닌 수**를 고릅니다.
 2. 고른 수에서 1을 뺍니다.
4. 3번 행동을 통해 배열을 팰린드롬으로 만들면 승리합니다.
5. 팰린드롬이 되지 않았다면 상대방에게 턴이 넘어가며, 둘 중 한 명이 승리할 때까지 2~4번을 반복합니다.

팰린드롬 게임을 여러 번 진행할 때, 각 게임의 결과를 구해야 합니다. **단, 두 사람 모두 최적으로 플레이한다고 가정합니다.** 즉, 양 플레이어는 이기기 위해 최선을 다하며, 실수를 하지 않습니다.

팰린드롬 게임을 시작할 때 주어지는 초기 배열을 담은 2차원 정수 배열 `queries` 가 매개변수로 주어질 때, 각 게임의 결과를 순서대로 배열에 담아 return 하도록 solution 함수를 완성해주세요.

첫 번째 플레이어가 승리한다면 1, 패배한다면 0을 배열에 담으면 됩니다.

제한사항

- $1 \leq \text{queries}$ 의 길이 ≤ 100

제한사항

- $1 \leq \text{queries}$ 의 길이 ≤ 100
 - `queries` 의 원소는 팰린드롬 게임의 초기 배열 을 나타냅니다.
 - $2 \leq \text{초기 배열}$ 의 길이 ≤ 5
 - $0 \leq \text{초기 배열}$ 의 원소 ≤ 9
 - 초기 배열 은 팰린드롬이 아닙니다.
 - 모든 초기 배열 의 길이는 동일합니다.

입출력 예

queries	result
[[2, 0], [3, 1]]	[0, 0]
[[1, 4, 3], [1, 2, 2]]	[0, 1]
[[0, 2, 0, 1], [0, 1, 0, 1]]	[1, 0]

입출력 예 설명

입출력 예 #1

총 두 번의 게임을 진행합니다.

첫 번째 게임의 초기 배열은 [2, 0] 입니다.

첫 번째 플레이어는 배열의 첫 번째 수에서 1을 빼는 행동만 할 수 있으며, 배열은 [1, 0] 이 됩니다. 두 번째 플레이어가 첫 번째 수에서 1을 빼면 배열은 [0, 0] 이 되고, 배열이 팰린드롬이기 때문에 두 번째 플레이어가 승리합니다. 따라서 0 을 정답 배열에 추가합니다.

두 번째 게임의 초기 배열은 [3, 1] 입니다.

첫 번째 플레이어가 어느 수를 고르든, 다음 턴에 두 번째 플레이어가 다른 수를 고르면 두 번째 플레이어가 무조건 승리합니다.

다시 말해, 첫 번째 플레이어가 만들 수 있는 배열은 [2, 1] 혹은 [3, 0] 인데, 두 경우 모두 두 번째 플레이어가 배열을 [2, 0] 으로 만들 수 있고, 이는 첫 번째 게임과 동일한 형태입니다. 따라서 두 번째 플레이어가 승리하며, 0 을 정답 배열에 추가합니다.

따라서 정답으로 [0, 0]을 return 해야 합니다.

입출력 예 #2

총 두 번의 게임을 진행합니다.

첫 번째 게임의 초기 배열은 [1, 4, 3] 입니다.

첫 번째 플레이어가 어떻게 플레이하더라도 [1, 1, 3] , [0, 0, 3] , [1, 0, 2] , [0, 4, 3] ... 등 항상 패배하는 형태밖에 만들 수 없습니다. 따라서 0 을 정답 배열에 추가합니다.

두 번째 게임의 초기 배열은 [1, 2, 2] 입니다.

첫 번째 플레이어가 배열을 [1, 2, 1] 로 만들면 바로 승리합니다. 따라서 1 을 정답 배열에 추가합니다.

따라서 정답으로 [0, 1]을 return 해야 합니다.

입출력 예 #3

총 두 번의 게임을 진행합니다.

첫 번째 게임의 초기 배열은 [0, 2, 0, 1] 입니다.

첫 번째 플레이어가 배열을 [0, 1, 0, 1] 로 만들면 두 번째 플레이어는 [0, 0, 0, 1] 혹은 [0, 1, 0, 0] 으로만 만들 수 있습니다. 다시 첫 번째 플레이어가 [0, 0, 0, 0] 을 만들면 승리합니다. 따라서 1 을 정답 배열에 추가합니다.

두 번째 게임의 초기 배열은 [0, 1, 0, 1] 입니다.

문제 설명

회사에서 직원들의 건강을 위해 운동복을 구입해주려고 합니다. 회사와 거래하는 의류 매장에서는 5가지 색상의 운동복을 판매하며, 편의를 위해 그 색상들은 아래와 같이 알파벳 앞글자를 사용하여 나타냅니다.

B=Black=검은색, W=White=흰색, R=Red=빨간색, Y=Yellow=노란색, G=Green=녹색

직원들이 원하는 운동복의 색상은 각각 다를 수 있기 때문에, 상. 하의 별로 선호하는 색상을 적어내도록 한 후, 의류 매장에 주문하기로 했습니다. 아래는 4명의 직원들이 적어낸, 선호하는 운동복의 색상입니다. 처음 알파벳은 상의의 색상, 두 번째 알파벳은 하의의 색상을 나타냅니다.

RG, WR, BW, GG

위의 예에서 4명의 직원들은 각각 **빨간색 상의 + 녹색 하의**, **흰색 상의 + 빨간색 하의**, **검은색 상의 + 흰색 하의**, **녹색 상의 + 녹색 하의**를 원하고 있습니다.

의류 매장에서 판매되는 운동복은 상. 하의가 같은 색으로 구성된 세트가 다른 색으로 구성된 세트보다 더 저렴합니다. 회사에서는 직접 재포장하여 직원들에게 나누어주는 수고를 감수하고서라도 비용을 아끼고 싶어 합니다.

예를 들어, 의류 매장에서 상. 하의가 같은 색인 운동복 세트를 5,000원에, 다른 색인 운동복 세트를 6,000원에 판매한다고 가정해보겠습니다. 4명의 직원이 원하는 운동복 색상이 **RG, WR, BW, GG** 라면, 상. 하의가 같은 색인 운동복 3세트 (**RR, WW, GG**)와 다른 색인 운동복 1세트 (**BG**)를 주문한 후, 직원들이 원하는 색으로 재포장하여 나누어 줄 수 있

줍니다. 이때, 발생하는 비용은 $5000 \times 3 + 6000 \times 1 = 21000$ 원입니다.

직원들이 원하는 운동복의 색상을 담은 문자열 배열 `color`, 운동복의 가격을 담은 정수형 배열 `prices`가 매개변수로 주어집니다. 필요한 경우, 회사가 직접 주문한 운동복을 재포장해서 직원들에게 원하는 색상으로 나누어 준다고 했을 때, 가장 저렴하게 구입할 수 있는 비용을 `return` 하도록 `solution` 함수를 완성해주세요.

제한사항

- `color`는 길이 1 이상 200,000 이하인 문자열 배열입니다.
 - `color`의 각 원소는 길이 2인 문자열로, `B`, `W`, `R`, `Y`, `G` 로만 이루어져 있습니다.
 - 각 문자열의 처음 알파벳은 상의의 색상, 두 번째 알파벳은 하의의 색상을 나타냅니다.
- `prices`는 길이 2인 정수형 배열입니다.
 - `prices`의 첫째 원소는 상, 하의가 같은 색인 운동복 세트의 가격, 두 번째 원소는 다른 색인 운동복 세트의 가격입니다.
 - 운동복의 가격은 1 이상 10,000 이하인 자연수이며, 첫째 원소는 두 번째 원소보다 작습니다.

입출력 예

color	prices	result
["RG", "WR", "BW", "GG"]	[5000, 6000]	21000
["RG", "WR", "BW", "GG"]	[2000, 6000]	10000
["BW", "RY", "BY"]	[9000, 10000]	30000

입출력 예 설명

입출력 예 #1

문제 예시와 같습니다.

입출력 예 #2

상. 하의 색상이 같은 운동복 세트 5개(RR, WW, GG, GG, BB)를 주문하면, 5개의 상의(R, W, G, G, B)와 5개의 하의(R, W, G, G, B)를 얻을 수 있습니다.

- 이때 발생하는 비용은 $2000 \times 5 = 10000$ 원입니다.

이것들을 재포장하여 직원들이 원하는 색상의 운동복 4세트(RG, WR, BW, GG)를 만들려면 4개의 상의(R, W, B, G)와 4개의 하의(G, R, W, G)가 필요합니다.

상의 하나(G)와 하의 하나(B)가 사용되지 않고 남게 됩니다. 옷이 남아도 괜찮습니다.

입출력 예 #3

상. 하의 색상이 다른 운동복 세트 3개(BW, RY, BY)를 주문하며 $10000 \times 3 = 30000$ 원의 비용을 지불하는 것이 가장 저렴하게 주문하는 방법입니다.

입출력 예 #4

상. 하의 색상이 같은 운동복 세트 2개(YY, WW)와 다른 색상의 운동복 세트 2개(RG, BW)를 주문한 후, 재포장하여 직원들에게 원하는 색상을 나누어주는 것이 가장 비용을 아낄 수 있는 방법입니다. 이때 발생하는 비용은 $7561 \times 2 + 8945 \times 2 = 33012$ 원입니다.

문제 설명

코드 품질 향상을 위해, 코드 리뷰를 실시하고자 합니다. 모든 개발자는 자신과 같은 프로젝트를 수행하는 동료들 중, n 명에게 코드 리뷰를 받아야 합니다. 단, 코드 리뷰에 너무 많은 시간을 투자하면 업무에 지장이 있을 수 있으므로, 한 명의 개발자가 리뷰할 수 있는 동료의 수는 최대 k 명으로 제한합니다.

프로젝트 1 : D, B, E, X

프로젝트 2 : A, C, E

프로젝트 3 : C, D, X

프로젝트 4 : A, B, G, X

위와 같이 알파벳 대문자로 표현된 7명(A, B, C, D, E, G, X)의 개발자가 4개의 프로젝트를 수행하고 있고, $n = 2$, $k = 3$ 인 경우, 다음과 같이 코드 리뷰어를 지정해줄 수 있습니다.

개발자	A	B	C	D	E	G	X
리뷰 어지 정방 법 1	B,C	A,D	A,D	B,C	A,C	B,X	D,
리뷰 어지 정방 법 2	E,C	D,E	E,A	X,B	D,B	A,B	C,
기타							

- 취소선이 그어진 방법 3은 A가 4명의 코드를 리뷰하기로 지정되었으므로, 잘못된 방법입니다.
 - $k = 3$ 이므로, 한 명이 리뷰할 수 있는 최대 동료의 수는 3명입니다.
- 취소선이 그어진 방법 4는 **B와 같은 프로젝트를 수행하고 있지 않는 C**가 B의 코드를 리뷰하기로 지정되었습니다. 따라서 잘못된 방법입니다.
- 취소선이 그어진 방법 5는 X의 리뷰어로 두 명이 아닌 한 명만 지정되었습니다. 뿐만 아니라, B의 리뷰어로 서로 다른 두 명이 지정되어야 하는데, D만 두 번 지정되었습니다. 따라서 잘못된 방법입니다.
 - $n = 2$ 이므로, 모든 개발자는 같은 프로젝트를 수행하고 있는 n 명의 개발자들에게 코드 리뷰를 받아야 합니다.
- 방법 1, 2는 모두 올바른 방법입니다. 이 중에서, 알파벳 순서가 빠른 개발자부터 자신의 코드를 리뷰할 동료들을 차례대로 이어 붙여서 문자열을 만들었을 때, 사전 순으로 빠른 방법을 정답으로 간주합니다. 즉, 올바른 **리뷰어 지정 방법** 을 나타내는 행에서, 쉼표를 제외하고 알파벳만 차례대로 이어 붙였을 때 사전 순으로 빠른 것을 정답으로 간주합니다.
 - 리뷰어 지정 방법 1을 나타내는 행에서 알파벳만 차례대로 이어 붙여서 만든 문자열은 "BCADADBCACBXDE"입니다.
 - 리뷰어 지정 방법 2를 나타내는 행에서 알파벳만 차례대로 이어 붙여서 만든 문자열은 "ECDEEAXBDBABCD"입니다.
 - "BCADADBCACBXDE" < "ECDEEAXBDBABCD" 이므로, 방법 1이 정답입니다.
- 방법 1, 2 말고도 주어진 조건에 맞게 코드 리뷰어를 지정하는 방법은 더 있습니다. 하지만, 사전 순으로 "BCADADBCACBXDE" 보다 빠른 방법으로 리뷰어를 할당하는 방법은 없습니다.

개발자들이 진행하고 있는 프로젝트를 나타내는 문자열 배열 `prj`, 각 개발자들이 리뷰를 받아야 하는 동료의 수를 나타내는 정수 `n`, 한 명의 개발자가 리뷰를 할 수 있는 최대 인원을 나타내는 정수 `k` 가 매개변수로 주어집니다. 알파벳 순서가 빠른 개발자부터 자신의 코드를 리뷰할 동료를 차례대로 이어 붙여서 문자열을 만들었을 때 사전 순으로 가장 빠른 문자열을 return 하도록 solution 함수를 완성해주세요.

제한사항

- $1 \leq \text{prj}$ 의 길이(프로젝트의 개수) ≤ 30
- $1 \leq \text{prj}$ 의 원소의 길이 ≤ 7
 - 즉, 한 프로젝트에 최대 7명의 개발자가 포함될 수 있습니다.
 - `prj`의 원소는 알파벳 대문자로만 이루어진 문자열입니다.
 - `prj`의 원소는 같은 알파벳을 2개 이상 포함하고 있지 않습니다.
 - 예를 들어, "AABC", "GHWGGH" 등은 `prj`의 원소가 될 수 없습니다.
- $2 \leq \text{prj}$ 에서 나타날 수 있는 서로 다른 알파벳의 개수 ≤ 7
 - 즉, 전체 개발자는 최소 2명에서 최대 7명입니다.
- $1 \leq n \leq 3$
- $1 \leq k \leq 3$
- 문제의 조건에 맞게 코드 리뷰어를 지정하는 방법이 있는 경우만 입력으로 주어집니다.

입출력 예 #1

각 라운드별 점수와 참가자가 선택한 모양, 점수를 가져가는 그룹을 나타내면 아래 표와 같습니다.

라운드	점수	참가자 1	참가자 2	참가자 3	참가자 4	참가자 5	참가자 6	점수를 가져가는 그룹
1	5	P	S	P	R	S	S	P
2	-2	S	S	R	R	S	S	S
3	1	R	R	R	R	R	R	-
4	3	R	R	S	S	P	P	-
5	-5	S	S	S	R	R	P	S

- 첫 번째 라운드에 걸린 점수는 양수(5)입니다. 세 가지 모양이 모두 나왔고, 그룹 크기가 모두 다르기 때문에 그룹 크기가 가장 큰 S를 우선 제외합니다. 남은 P와 R 중에서 가위바위보를 이기는 그룹 P가 점수를 가져갑니다. 따라서 참가자 1과 참가자 3이 5점을 얻습니다.
- 두 번째 라운드에 걸린 점수는 음수(-2)입니다. 두 가지 모양만 나왔기 때문에 두 그룹 중 가위바위보를 지는 그룹 S가 점수를 가져갑니다. 따라서 참가자 1, 참가자 2, 참가자 5, 참가자 6이 -2점을 얻습니다.
- 세 번째 라운드에 걸린 점수는 양수(1)입니다. 한 가지 모양만 나왔기 때문에 이번 라운드의 점수를 다음 라운드에 합치고, 다음 라운드로 넘어갑니다.
- 네 번째 라운드에 걸린 점수는 이전 라운드의 점수 1을 합한 4입니다. 세 가지 모양이 나왔고 모든 그룹의 크기가 같으므로, 이번 라운드의 점수를 다음 라운드에 합치고, 다음 라운드로 넘어갑니다.
- 다섯 번째 라운드에 걸린 점수는 이전 라운드의 점수 4를 합한 -1입니다. 세 가지 모양이 모두 나왔고, 그룹 크기가 모두 다르기 때문에 그룹 크기가 가장 작은 P를 우선 제외합니다. 남은 R과 S 중에서 가위바위보를 지는 그룹 S가 점수를 가져갑니다. 따라서 참가자 1, 참가자 2, 참가자 3이 -1점을 얻습니다.

모든 라운드가 끝나고 각 참가자가 얻은 점수의 합은 순서대로 [2, -3, 4, 0, -2, -2]입니다. 따라서 최고득점자의 점수인 4를 return 합니다.