

# Week 4

- 주의 사항: 부정행위 금지, STL 사용금지(string 사용가능), 배열 또는 연결 리스트를 이용한 **큐를 구현**하여 문제를 해결할 것.
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

## 문제 2

두 명의 플레이어는 동일한 개수의 카드덱을 받는다. 각 캐릭터 카드는 체력이 있는데, 덱 순서대로 한 장씩 뽑아 대결을 한다. 여기서 대결이란 두 카드 사이에 체력이 많은 쪽의 체력에서 적은 쪽의 체력만큼을 빼는 것이며, 체력이 많은 쪽이 해당 대결의 승자이다.

덱 순서대로 1:1로 대결 후 이긴 캐릭터를 갖고 있던 플레이어는 해당 대결에서 승점 1점을 얻으며, **대결 후에 남은 캐릭터의 체력-1은 다음에 대결할 상대 캐릭터에 추가된다.** 모든 대결이 끝난 후 승점이 높은 플레이어가 최종 승리한다.

예를 들어 두 플레이어 P1, P2에게 각각 6개의 캐릭터 카드가 포함된 덱들이 다음 R0와 같이 주어지고, R1부터 R6까지 6라운드의 대결을 한다고 하자.

R1에서는 P1의 첫번째 캐릭터 카드의 12와 P2의 첫번째 캐릭터 카드의 10이 대결을 진행한다. 이 때, P1의 캐릭터가 체력이 2가 더 많으므로 P1은 승점을 1점 획득하고, P1의 첫번째 캐릭터의 (남은 체력-1)인 1이 상대의 두번째 카드인 11에 추가되어 **R2에서는 12로 적용된 후 대결이 진행된다.** 이와 같은 방법으로 모든 덱에 대하여 실행한다. 단, 비겼을 경우 승점에는 변동이 없으며, 다음 대결로 넘어가는 체력은 없다.

	Ver 1
R0	P1 : 12 11 7 5 3 1 P2 : 10 11 13 6 4 2
R1	P1 : <b>12</b> 11 7 5 3 1 // 12 > 10, P1 승점+1 P2 : <b>10</b> 11 13 6 4 2 // P1 : P2 = 1 : 0
R2	P1 : <b>11</b> 7 5 3 1 // <b>12=11+(2-1)</b> , 11 < 12, P2 승점+1 P2 : <b>12</b> 13 6 4 2 // P1 : P2 = 1 : 1
R3	P1 : <b>7</b> 5 3 1 // <b>7=7+(1-1)</b> , 7 < 13, P2 승점+1 P2 : <b>13</b> 6 4 2 // P1 : P2 = 1 : 2
R4	P1 : <b>10</b> 3 1 // <b>10=5+(6-1)</b> , 10 > 1, P1 승점+1 P2 : <b>6</b> 4 2 // P1 : P2 = 2 : 2
R5	P1 : <b>3</b> 1 // <b>7=4+(4-1)</b> , 3 < 7, P2 승점+1 P2 : <b>7</b> 2 // P1 : P2 = 2 : 3
R6	P1 : <b>4</b> // <b>4=1+(4-1)</b> , 4 > 2, P1 승점+1 P2 : <b>2</b> // P1 : P2 = 3 : 3
결과	DRAW

# 입력

- 1. 첫 번째 줄에 테스트 케이스의 개수  $T$ 가 주어진다. ( $1 \leq T \leq 100,000$ ).
- 2. 각 케이스마다 다음이 반복되어 나타난다. ( $1 \leq N \leq 20, 1 \leq x_i, y_i \leq 20, 1 \leq i \leq N$ )
  - 1) 카드의 개수  $N$ 이 주어진다.
  - 2) P1의  $N$ 개의 캐릭터 체력  $x_i$ 가 띄어쓰기로 구분되어 주어진다.
  - 3) P2의  $N$ 개의 캐릭터 체력  $y_i$ 가 띄어쓰기로 구분되어 주어진다.

# 출력

각 테스트 케이스마다 최종 승자를 출력한다. P1이 승자라면 1, P2가 승자라면 2, 비긴 경우는 0이다.

## 예제 입출력

예제 입력	예제 출력
2 6 12 11 7 5 3 1 10 11 13 6 4 2 3 4 3 2 3 2 1	0 1