**문제 설명**

철수는 해외여행 자금 마련을 위해 아르바이트를 하려고 합니다. 모든 아르바이트는 시작 날짜와 끝 날짜가 정해져 있으며, 급여도 제각각입니다. 철수는 최대한 많은 돈을 벌기 위해 아르바이트를 여러 개 하려고 합니다. 단, 아르바이트를 동시에 두 개 이상 할 수는 없으며, 아르바이트가 끝나는 날 다른 아르바이트를 시작하는 것은 가능합니다.

각 아르바이트의 시작 날짜와 끝 날짜, 급여가 담긴 배열 part\_times가 매개변수로 주어질 때, 철수가 최대로 벌 수 있는 돈은 얼마인지 return 하도록 solution 함수를 완성해주세요.

제한사항

* part\_times는 2차원 배열입니다.
* part\_times의 세로(행) 길이는 1 이상 15 이하입니다.
* part\_times의 가로(열) 길이는 3입니다.
* part\_times의 원소는 [S, E, C] 형태입니다.
  + S는 아르바이트 시작일, E는 아르바이트 종료일, C는 해당 아르바이트의 급여를 나타냅니다.
  + S는 항상 E보다 작습니다.
  + S와 E는 1 이상 100 이하인 자연수입니다.
  + C는 1 이상 1,000 이하인 자연수입니다.

입출력 예

| **part\_times** | **result** |
| --- | --- |
| [[3, 6, 3], [2, 4, 2], [10, 12, 8], [11, 15, 5], [1, 8, 10], [12, 13, 1]] | 19 |
| [[1, 2, 1], [1, 2, 2], [2, 3, 1], [3, 4, 1], [1, 4, 2]] | 4 |

입출력 예 설명

입출력 예 #1

각 아르바이트의 시작, 끝 날짜와 급여는 다음과 같습니다.

| **시작(일)** | **끝(일)** | **급여(원)** |
| --- | --- | --- |
| 3 | 6 | 3 |
| 2 | 4 | 2 |
| 10 | 12 | 8 |
| 11 | 15 | 5 |
| 1 | 8 | 10 |
| 12 | 13 | 1 |

예를 들어 첫 번째와 두 번째 아르바이트의 경우 3일, 4일 날짜가 겹치게 되므로 두 아르바이트를 동시에 할 수는 없습니다. 그러나, 세 번째, 다섯 번째 아르바이트는 8일에 아르바이트가 종료된 후 10일에 다른 아르바이트를 다시 시작하기 때문에 두 아르바이트를 모두 할 수 있습니다.

위 예시에서는 세 번째, 다섯 번째, 여섯 번째 아르바이트를 하면 총 19원을 벌 수 있으며, 이때가 최대입니다.

입출력 예 #2

두 번째, 세 번째, 네 번째 아르바이트를 하면 총 4원을 벌 수 있으며, 이때가 최대입니다.

**문제 설명**

RPG 게임에 1부터 N까지 번호가 하나씩 붙은 스킬 N 종류가 있습니다. 이때, 각 스킬을 배우는데 필요한 스킬 포인트를 정하려 합니다.

스킬에는 다음과 같이 상위 스킬과 선행 스킬이라는 개념이 있습니다.

* 상위 스킬 : 어떤 스킬을 배웠을 때 배울 수 있는 바로 다음 스킬
  + 상위 스킬이 없는 최상위 스킬은 딱 하나만 있습니다.
  + 최상위 스킬을 제외한 **모든 스킬은 상위 스킬을 하나씩 가지고** 있습니다.
* 선행 스킬 : 어떤 스킬을 배우기 위해 먼저 배워야 하는 모든 스킬
  + 선행 스킬이 없는 경우 스킬을 바로 배울 수 있습니다.
  + 현재 스킬을 상위 스킬로 하는 모든 스킬 및 그 스킬들의 선행 스킬을 모두 포함합니다.
  + 선행 스킬을 모두 배워야 현재 스킬을 배울 수 있습니다.

모든 스킬을 익히는데 필요한 스킬 포인트를 total\_sp만큼으로 정했을 때, 각 스킬에 필요한 스킬 포인트를 다음 규칙에 따라 정하려 합니다.

* 어떤 스킬을 익히는데 필요한 스킬 포인트는 선행 스킬들을 익히는데 필요한 스킬 포인트의 합입니다.
* 선행 스킬이 없는 스킬들을 익히는데 필요한 스킬 포인트는 모두 같습니다.

총 스킬 포인트 total\_sp, 각 스킬의 상위 스킬을 담고 있는 배열 skills가 주어질 때, 각 스킬을 익히는데 필요한 스킬 포인트 배열을 return 하도록 solution 함수를 작성해주세요.

제한사항

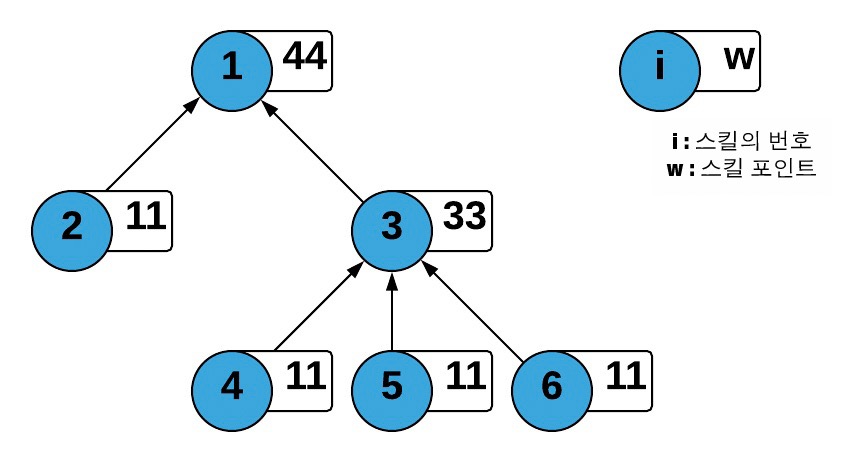
* 2 ≤ N ≤ 100000, N은 자연수
* 1 ≤ total\_sp ≤ 1,000,000,000, total\_sp는 자연수
  + total\_sp는 규칙에 맞게 정확히 분배할 수 있는 경우만 입력으로 주어집니다.
* 각 스킬을 익히는데 필요한 스킬 포인트는 자연수입니다.
* skills의 길이는 스킬 개수(N)보다 1 작습니다.
* skills의 원소는 [a, b]의 형태입니다.
  + a와 b는 스킬 번호를 나타냅니다.
  + (1 ≤ a, b ≤ N), (a ≠ b), a, b는 자연수
  + 스킬 a는 스킬 b의 상위 스킬입니다(즉, b → a의 형태)
  + 스킬 관계가 잘못 주어지는 경우는 없습니다.
* 배열의 첫 번째 원소부터 차례대로 스킬을 배우는데 필요한 스킬 포인트를 채워서 return 해주세요.
  + return 배열의 i - 1번째 원소가 i번 스킬을 배우는데 필요한 스킬 포인트이어야 합니다.

입출력 예

| **total\_sp** | **skills** | **return** |
| --- | --- | --- |
| 121 | [[1, 2], [1, 3], [3, 6], [3, 4], [3, 5]] | [44, 11, 33, 11, 11, 11] |

입출력 예 설명

예제 #1



위와 같이 스킬 포인트를 정하면 총 스킬 포인트가 121이 되며 조건을 만족합니다.