

- COS Pro 샘플 문제 중 그래프 응용 문제를 학습하는 날입니다.

Union\_find 알고리즘을 이용하기에 적절한 문제를 통해 유사한 문제에 적용할 수 있는 Union\_find 알고리즘을 익혀보세요.

제공된 문제는 빈칸 문제입니다. 빈칸을 채워 결과를 얻고 전체 코드를 작성하는 연습을 하세요.

제공 코드는 다음과 같습니다.

- 5 차 1 급 7\_initial\_code.py

먼저 문제를 스스로 풀어 보시고 솔루션 파일을 참고하세요.

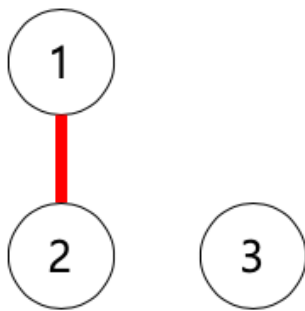
솔루션 코드는 다음과 같습니다.

- 5 차 1 급 7\_solution\_code.py

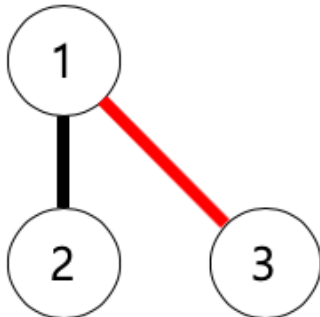
# [1] COS Pro 1급 5차 #문제7

그래프의 노드 수와 노드 연결 순서가 주어질 때, 몇 번째 연결에 사이클이 생기는지 알고 싶습니다. 예를 들어, 노드가 3개이고 노드를 [[1, 2], [1, 3], [2, 3]] 순으로 연결한다면 아래 그림과 같습니다.

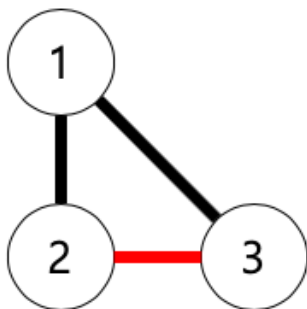
\* 1번째 연결



\* 2번째 연결



\* 3번째 연결



따라서 3번째 연결에서 사이클이 생깁니다.

그래프의 노드 수  $n$ , 노드 연결 순서 `connections`가 매개변수로 주어질 때, 몇 번째 연결에 사이클이 생기는지 `return` 하도록 `solution` 메소드를 작성하려 합니다. 빈칸을 채워 전체 코드를 완성해주세요.

#### #####매개변수 설명

그래프의 노드 수  $n$ , 노드 연결 순서 `connections`가 `solution` 메소드의 매개변수로 주어집니다.

\* 그래프의 노드 수  $n$ 은 3 이상 10 이하입니다.

\* `connections`은 길이가 3 이상 50 이하인 배열입니다.

\* `connections` 배열의 각 행은 `[a, b]`는  $a$ 번 노드와  $b$ 번 노드를 연결한다는 의미입니다.

\* 항상 사이클이 생기는 경우만 주어집니다.

#### #####return 값 설명

몇 번째 연결에서 사이클이 생기는지 `return` 합니다.

#### #####예제

n	connections	return
3	[[1, 2], [1, 3], [2, 3]]	3

#### #####예제 설명

문제에 나온 예와 같습니다.