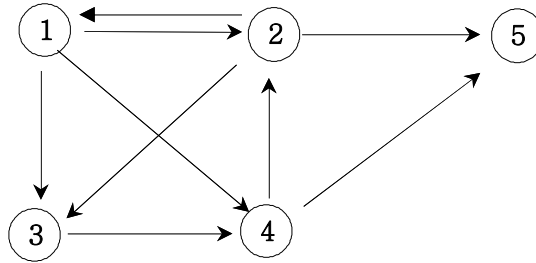


경로 탐색(인접행렬)

방향그래프가 주어지면 1번 정점에서 N번 정점으로 가는 모든 경로의 가지 수를 출력하는 프로그램을 작성하세요. 아래 그래프에서 1번 정점에서 5번 정점으로 가는 가지 수는

1 2 3 4 5
 1 2 5
 1 3 4 2 5
 1 3 4 5
 1 4 2 5
 1 4 5



총 6 가지입니다.

입력설명

매개변수 $n(1 \leq n \leq 20)$ 과 매개변수 `edge`에 간선의 연결정보가 주어진다.

출력설명

총 경로의 가지수를 반환한다.

입력예제 1

5, [[1, 2], [1, 3], [1, 4], [2, 1], [2, 3], [2, 5], [3, 4], [4, 2], [4, 5]]

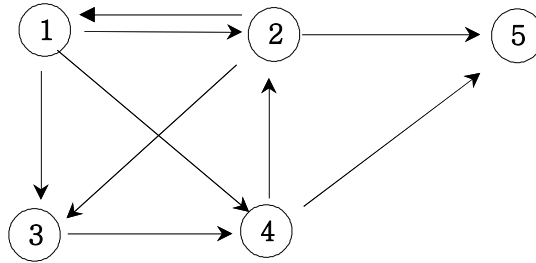
출력예제 1

6

경로 탐색(인접리스트)

방향그래프가 주어지면 1번 정점에서 N번 정점으로 가는 모든 경로의 가지 수를 출력하는 프로그램을 작성하세요. 아래 그래프에서 1번 정점에서 5번 정점으로 가는 가지 수는

1 2 3 4 5
 1 2 5
 1 3 4 2 5
 1 3 4 5
 1 4 2 5
 1 4 5



총 6 가지입니다.

입력설명

매개변수 $n(1 \leq n \leq 10,000)$ 과 매개변수 `edge`에 간선의 연결정보가 주어진다.

출력설명

총 경로의 가지수를 반환한다.

입력예제 1

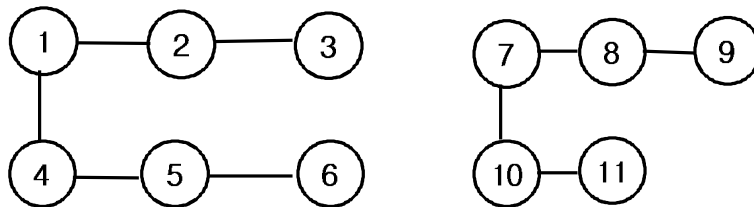
5, [[1, 2], [1, 3], [1, 4], [2, 1], [2, 3], [2, 5], [3, 4], [4, 2], [4, 5]]

출력예제 1

6

컴퓨터 개수

현수는 컴퓨터 학원에 다닌다. 현수가 다니는 컴퓨터 학원은 서버 컴퓨터가 있는 큰 교실과 수학을 공부하는 작은 교실로 되어 있다. 서버 컴퓨터가 있는 큰 교실의 모든 컴퓨터는 서버와 직간접적으로 연결되어 인터넷이 되지만 수학교실에 있는 컴퓨터들은 서버와 연결되어 있지 않아 인터넷은 되지 않는다.



서버 컴퓨터는 1번 컴퓨터이다. 1, 2, 3, 4, 5, 6번 컴퓨터는 인터넷이 되지만 7, 8, 9, 10, 11번 컴퓨터는 수학교실에 있는 컴퓨터들로 인터넷이 되지 않는다.

수학교실에 있는 컴퓨터들끼리는 서로 연결이 되어 있을 수도 있고, 연결이 되어 있지 않을 수도 있다.

현수가 다니는 학원의 수학교실에는 몇 대의 컴퓨터가 있는지 개수를 출력하는 프로그램을 작성하세요.

입력설명

매개변수 N에 현수가 다니는 컴퓨터 학원의 컴퓨터의 총개수 $N(5 \leq N \leq 200)$ 이 주어집니다. 매개변수 edge에 각 컴퓨터간의 연결정보가 주어집니다.

출력설명

수학교실에 있는 컴퓨터의 개수를 반환합니다.

입력예제 1

11, [[1, 2], [1, 4], [2, 3], [4, 5], [5, 6], [7, 8], [7, 10], [8, 9], [10, 11]]

출력예제 1

5

입력예제 2

12, [[1, 2], [1, 7], [1, 8], [1, 6], [8, 11], [11, 12]]

출력예제 2

5

동아리 개수

현수가 다니는 학교에는 동아리가 많이 있습니다. 현수가 다니는 학교의 동아리의 개수를 구하는 프로그램을 작성하세요.

현수가 다니는 학교의 학생은 1번부터 N번까지 번호가 부여된 N명의 학생들로 되어 입력됩니다. 만약 1번 학생과 2번 학생이 같은 동아리 이면 (1, 2) 순서쌍으로 입력되며, (2, 3)이 입력되면 1, 2, 3번 학생은 같은 동아리입니다.

모든 학생은 동아리를 가지고 있습니다.

■ 입력설명

매개변수 n에 현수가 다니는 학교의 학생수 $N(5 \leq N \leq 30,000)$ 이 주어집니다.

매개변수 edges에 두 학생의 정보를 나타내는 순서쌍의 주어집니다.

■ 출력설명

현수가 다니는 학교의 동아리 개수를 반환하세요.

■ 매개변수 형식 1

7, [[1, 2], [2, 3], [1, 4], [1, 5]]

■ 반환값 형식 1

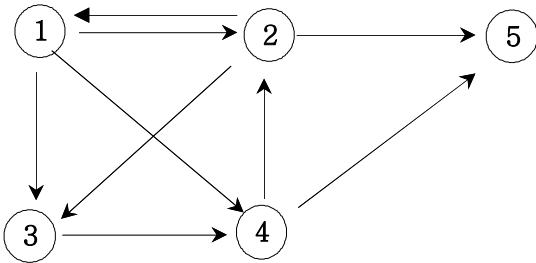
3

출력설명 : (1, 2, 3, 4, 5), (6), (7) 3개의 동아리가 존재합니다.

즉 1번, 2번, 3번, 4번, 5번 학생이 같은 동아리 이고, 6번 혼자서 동아리, 7번 혼자서 동아리가 됩니다.

경로탐색(인접행렬)

	1	2	3	4	5
ch					



```
def DFS(v, n, ch, graph):
    global answer
    if v == n:
        answer += 1
    else:
        for i in range(n+1):
            if graph[v][i] == 1 and ch[i] == 0:
                ch[i] = 1
                DFS(i, n, ch, graph)
                ch[i] = 0
```