

Day 13

BFS/DFS, 다익스트라 알고리즘


취업노하우

주중 C 듣고

화/목 - C 자료구조 과정 개설 예정

주말 코테 계속 듣기

예시 문제:

 DFS / BFS Problems.ipynb

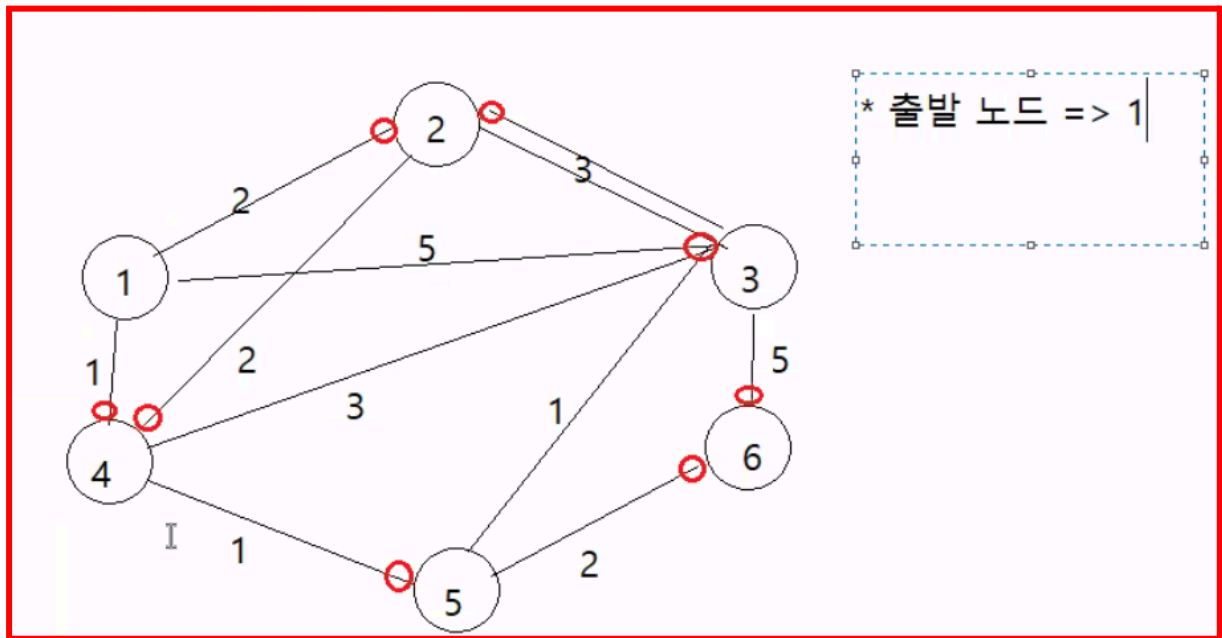
DFS / BFS 둘다 풀이 가능:

<https://school.programmers.co.kr/learn/courses/30/lessons/43162>

다익스트라 알고리즘

- 최단 경로 알고리즘이다. 특정 노드에서 출발하여 다른 노드로 가는 각각의 최단 경로를 구하는 알고리즘이다.(음의 간선(X))
- 즉 비용이 가장 적은 노드를 선택해서 임의의 과정을 반복한다.
 1. 출발 노드를 설정
 2. 최단 거리 테이블을 초기화
 3. 방문하지 않은 노드중에서 최단 거리가 가장 짧은 노드 선택
 4. 해당 노드를 거쳐 다른 노드로 가는 비용을 계산하여 테이블 갱신 (그리드 알고리즘 이랑 같다)
 5. 3, 4번 과정을 반복한다.

ex) 다익스트라 알고리즘



- 다익스트라는 어디에서 출발하는지 알려준다.
- * 출발 노드는 가장 짧은 노드를 일반적으로 선택 (출발노드로의 거리는 0이다)

거리표를 작성하면 이해하기 쉽다:

노드	1	2	3	4	5	6
거리	0	inf	inf	inf	inf	inf

* 무한대 값 표시

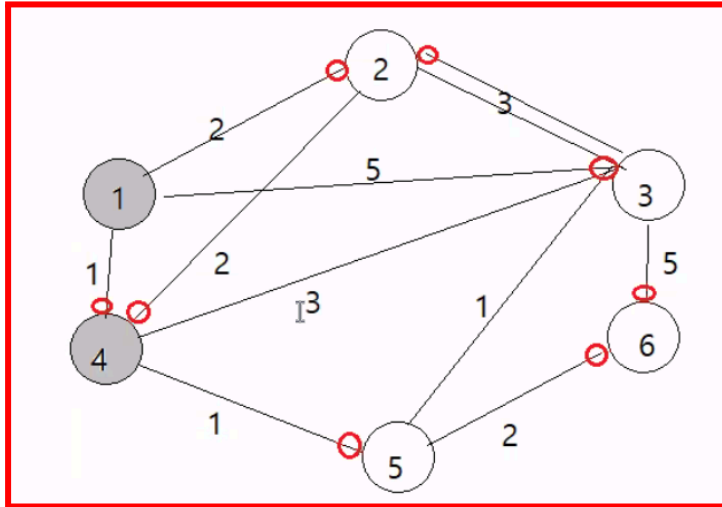
1. 999,999,999
2. `int(1e9)`

1번 노드가 시작노드이고 연결(인접)되어있는 간선을 하나씩 확인한다:

노드	1	2	3	4	5	6
거리	0	2	5	1	inf	inf

다음으로는 더 효율적으로 가는 루트가 있는지 (가장 짧은 거리로) 찾는다.

1. 1번노드를 기준으로 가장 가중치가 작은 노드를 찾는다.
2. 그 노드를 통해서 갈 수 있는 노드를 찾는다.

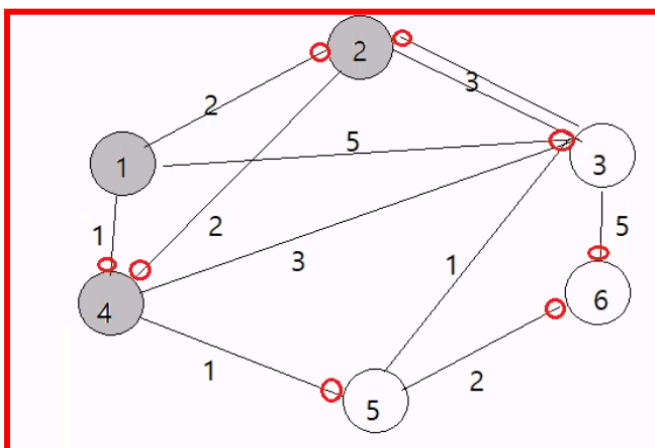


$1 \rightarrow 4 \rightarrow 3 \Rightarrow$ 가중치 : $1 + 3 = 4$

$1 \rightarrow 4 \rightarrow 5 \Rightarrow$ 가중치 : $1 + 1 = 2$

노드	1	2	3	4	5	6
거리	0	2	4	1	2	inf

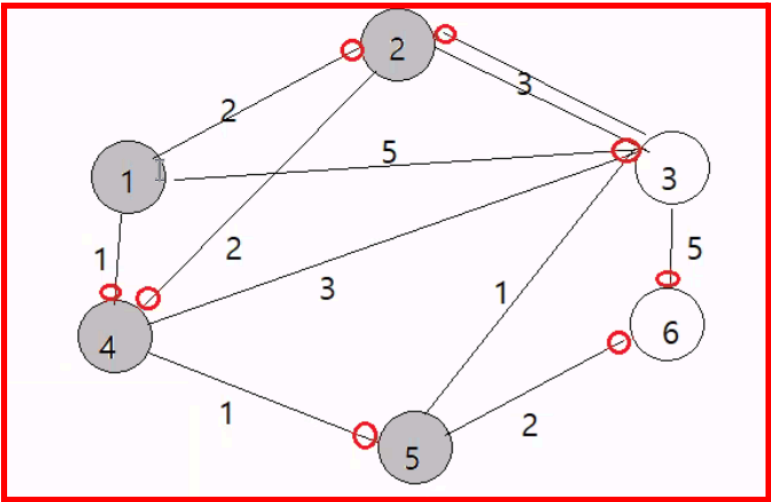
그 다음으로 가장 작은 노드는 2이다 (가중치 2):



하지만 노드 2를 통하는 길 중에서 더 효율적인 루트는 없다.

노드	1	2	3	4	5	6
거리	0	2	4	1	2	inf

그 다음으로 가장 작은 노드는 5이다. (가중치 2)

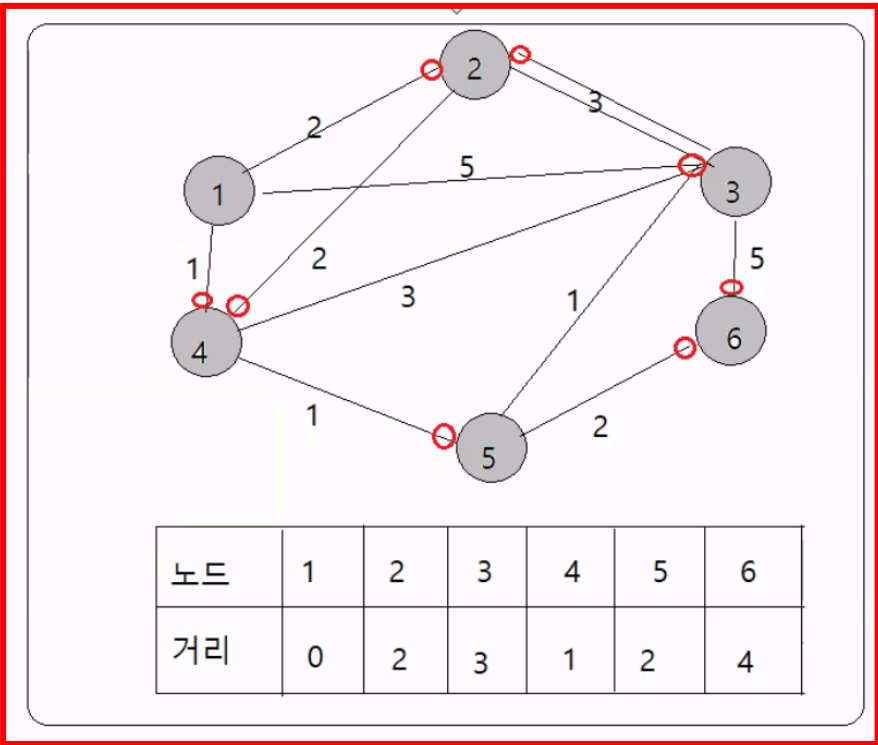


1 → 4 → 5 → 6 ⇒ 가중치 : 4

1 → 4 → 5 → 3 ⇒ 가중치 : 3

노드	1	2	3	4	5	6
거리	0	2	3	1	2	4

그 다음으로 가장 작은 노드는 3 → 6 순서이다.



하지만 더 효율적인 루트는 만나온다. 따라서 최종 표가 완성된다.

* 최대 마을의 개수를 함수를 작성

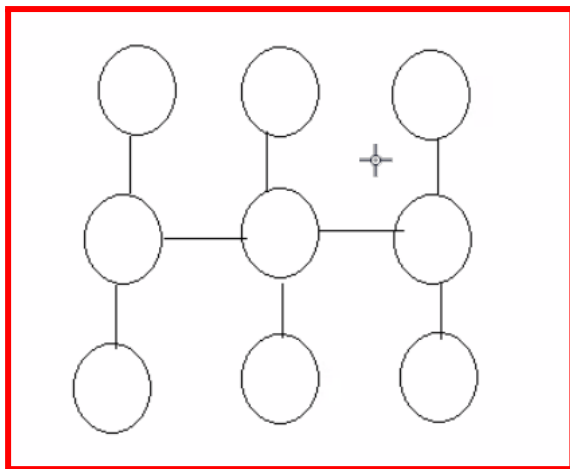
- 다익스트라 알고리즘 (heap)
 - 최대값 최소값 구할때 사요

문제 풀어보기 (2번 다익스트라 알고리즘 문제):

DFS / BFS Problems.ipynb

3번 문제:

<https://school.programmers.co.kr/learn/courses/30/lessons/86971>



- 이 구조에서 트리화 시킬 수 있는 구조를 찾아야 한다.
- 어느 노드를 잘라서 2진 트리화 시킬 수 있는지가 중요하다.
- 트리를 만들어주는게 중요하다.
- 2개의 트리로 만들어서 노드의 개수를 구한다음 그 차이를 최소값으로 만드는것.