

자료구조론 실습

Shortest Path

2017. 06. 09

한양대학교

이주홍

Assignment 9 - Dijkstra Algorithm

- $n(1 \leq n \leq 1,000)$ 개의 도시가 있다.
- 한 도시에서 출발하여 다른 도시에 도착하는 $m(1 \leq m \leq 100,000)$ 개의 버스가 있다.
- A번째 도시에서 B번째 도시 까지 가는데 드는 최소비용을 출력하여라.

Assignment 9 – Dijkstra Algorithm

- **입출력 설명**

- 첫째 줄에 도시의 개수 n ($1 \leq n \leq 1,000$)이 주어진다.
- 둘째 줄에 버스의 개수 m ($1 \leq m \leq 100,000$)이 주어진다.
- 셋째 줄부터 $m+2$ 째 줄까지 m 개의 버스의 정보가 주어진다.
 - 한줄에 u, v, w 3개의 정수가 주어진다.
 - u : 시작 도시
 - v : 도착 도시
 - w : 버스 비용, (0보다 크거나 같고 100,000보다 작은 정수)
- $m+3$ 째 줄에는 우리가 구하고자하는 출발 도시와 도착 도시의 번호가 주어진다.
- 출발 도시에서 도착 도시까지 가는데 드는 최소 비용을 출력한다.
- 출발 도시에서 도착 도시까지 갈 수 있는 경우만 주어진다.

Assignment 9 - Dijkstra Algorithm

입력 예시

5
8

도시의 개수 5개 (1~5)
도시를 연결하는 버스의 개수 8개

m(=8)개의 버스 정보

1 2 2
1 3 3
1 4 6
1 5 10
2 4 3
3 4 1
3 5 4
4 5 1
1 5

1에서 5로 가는 최소 비용을 구하라

출력 예시

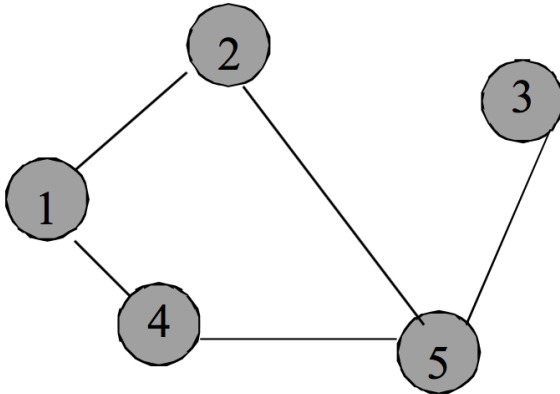
5

1에서 5로 가는
최소비용 = 5

Graph Representation

Adjacency Matrix

- 0/1 $n \times n$ matrix, where $n = \#$ of vertices
- $A(i,j) = 1$ iff (i,j) is an edge



	1	2	3	4	5
1	0	1	0	1	0
2	1	0	0	0	1
3	0	0	0	0	1
4	1	0	0	0	1
5	0	1	1	1	0

Graph Representation

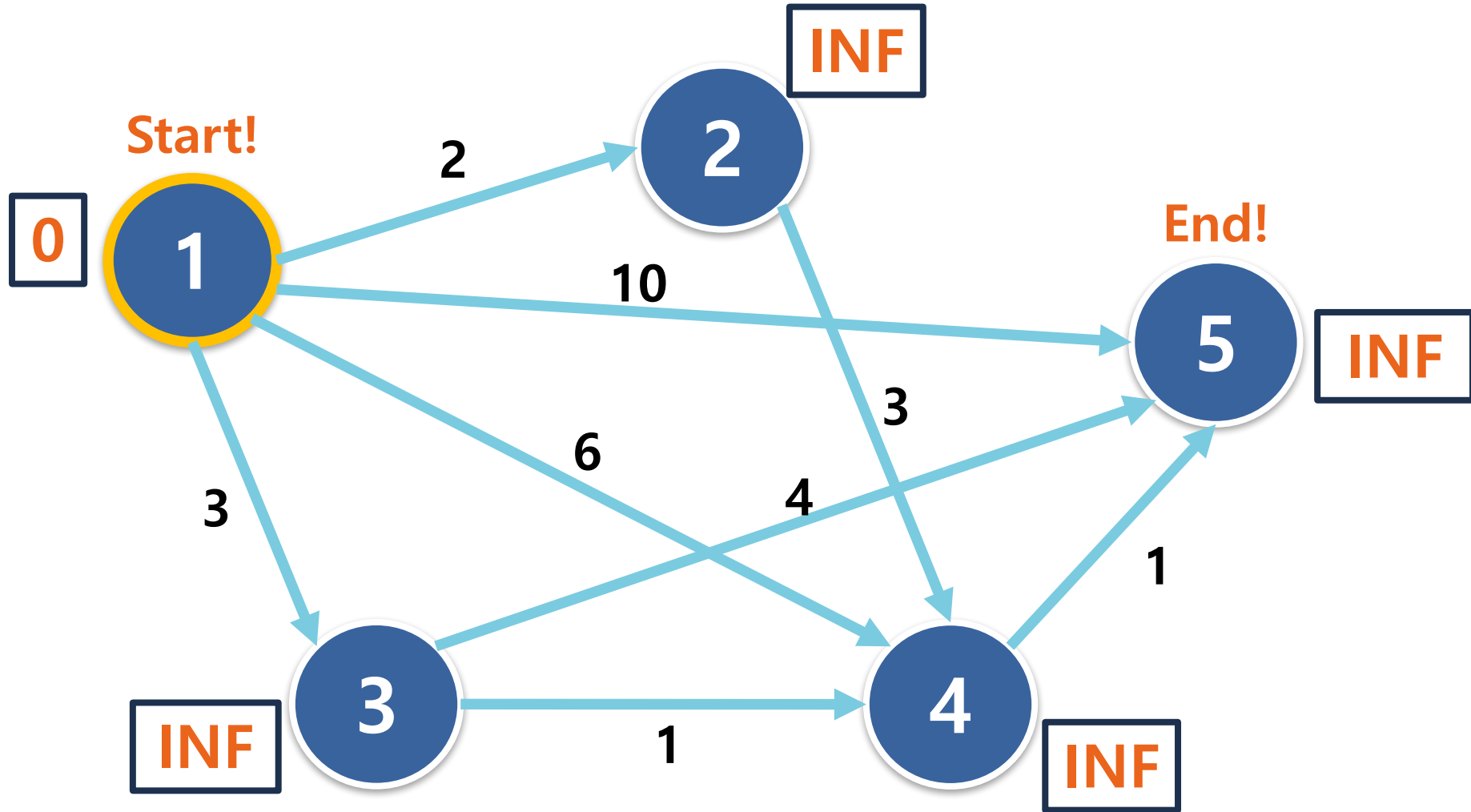
```
int u, v;  
scanf("%d %d %d", &u, &v, &w);  
node[u][v] = w;
```



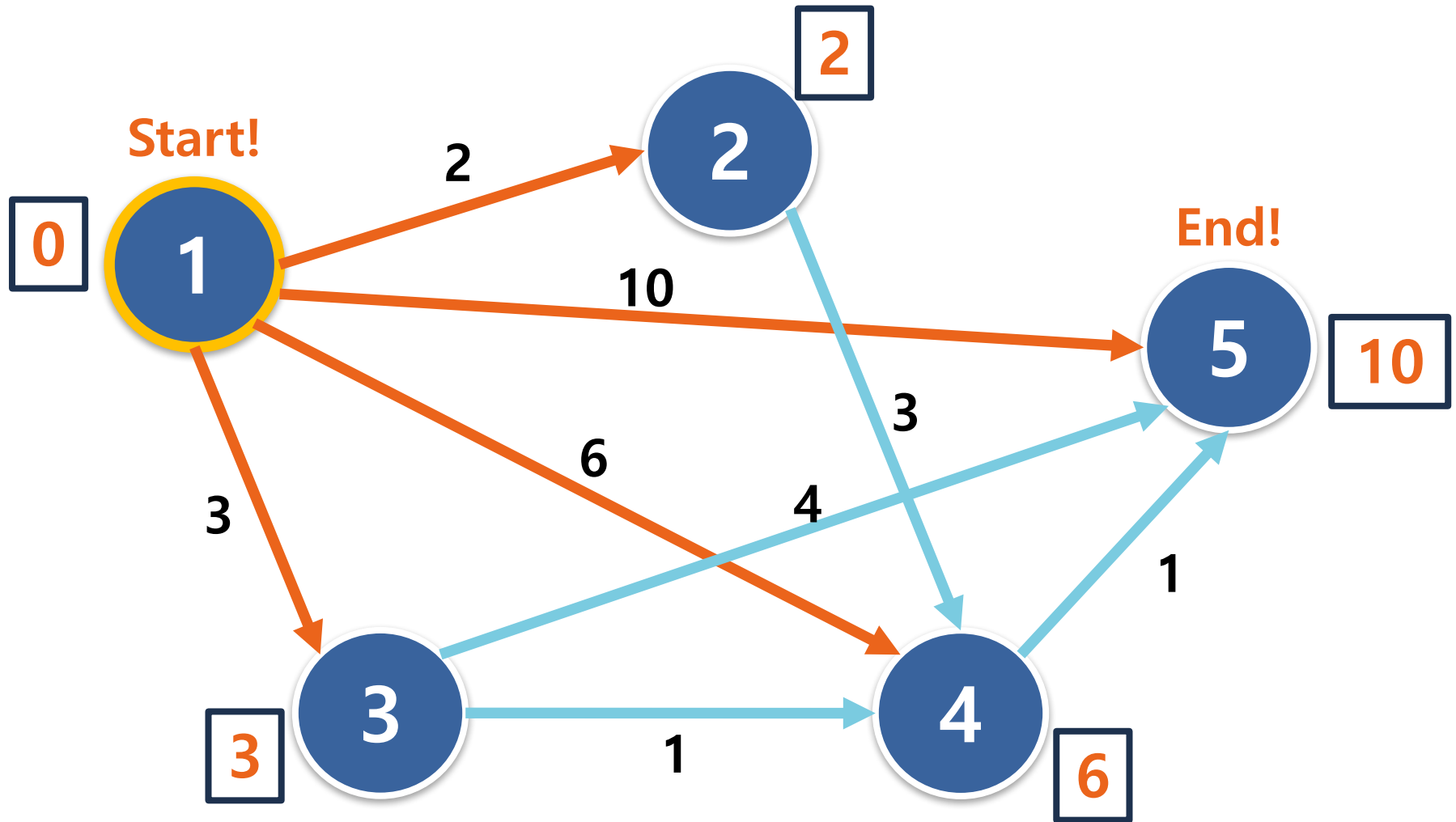
Dijkstra Algorithm

1. 가장 가까운 노드를 선택한다.
2. 해당 노드를 기준으로
각 노드까지의 Distance를 갱신한다.
3. 반복(n번)

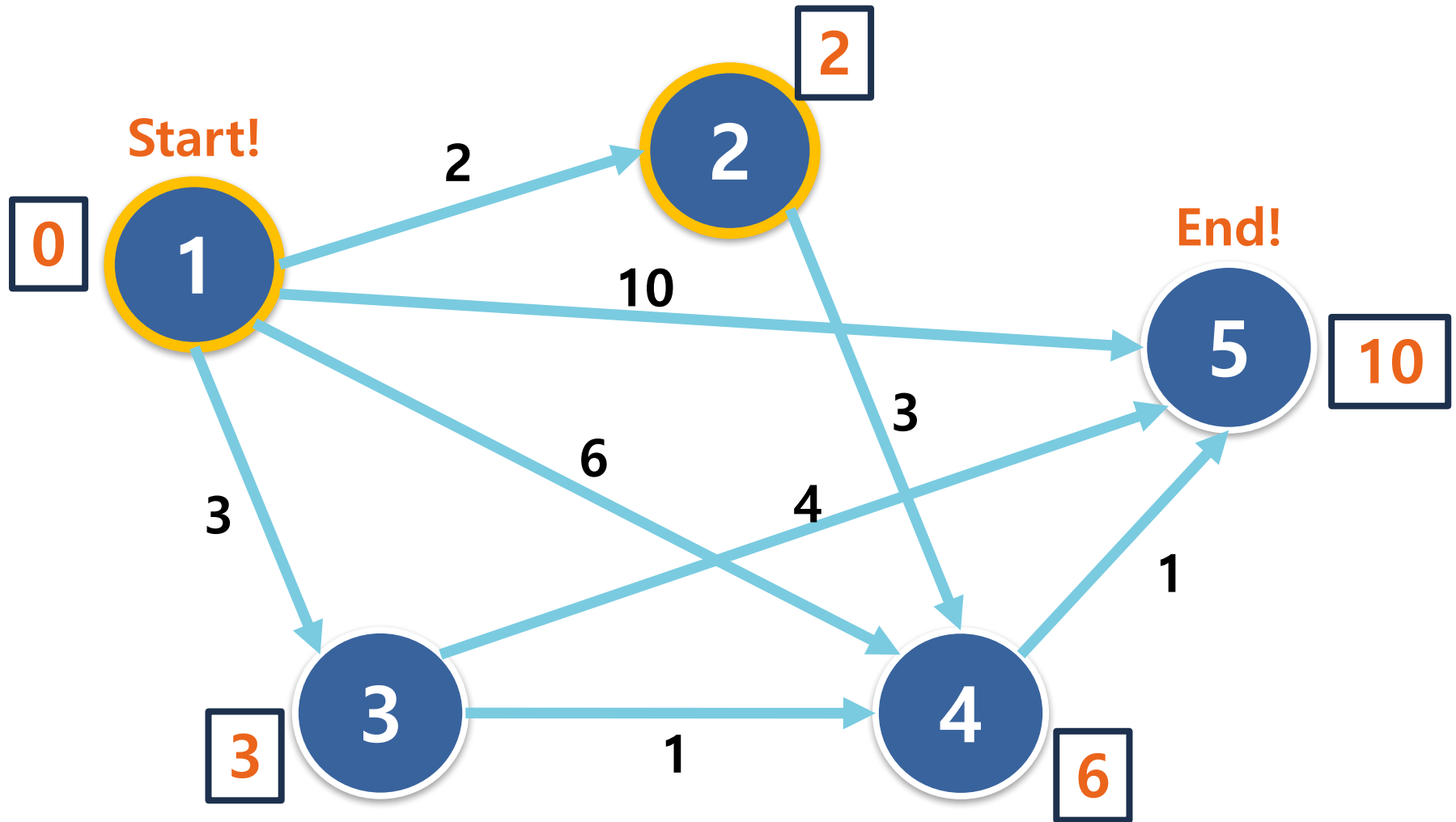
Dijkstra Algorithm - min 노드(1) 선택



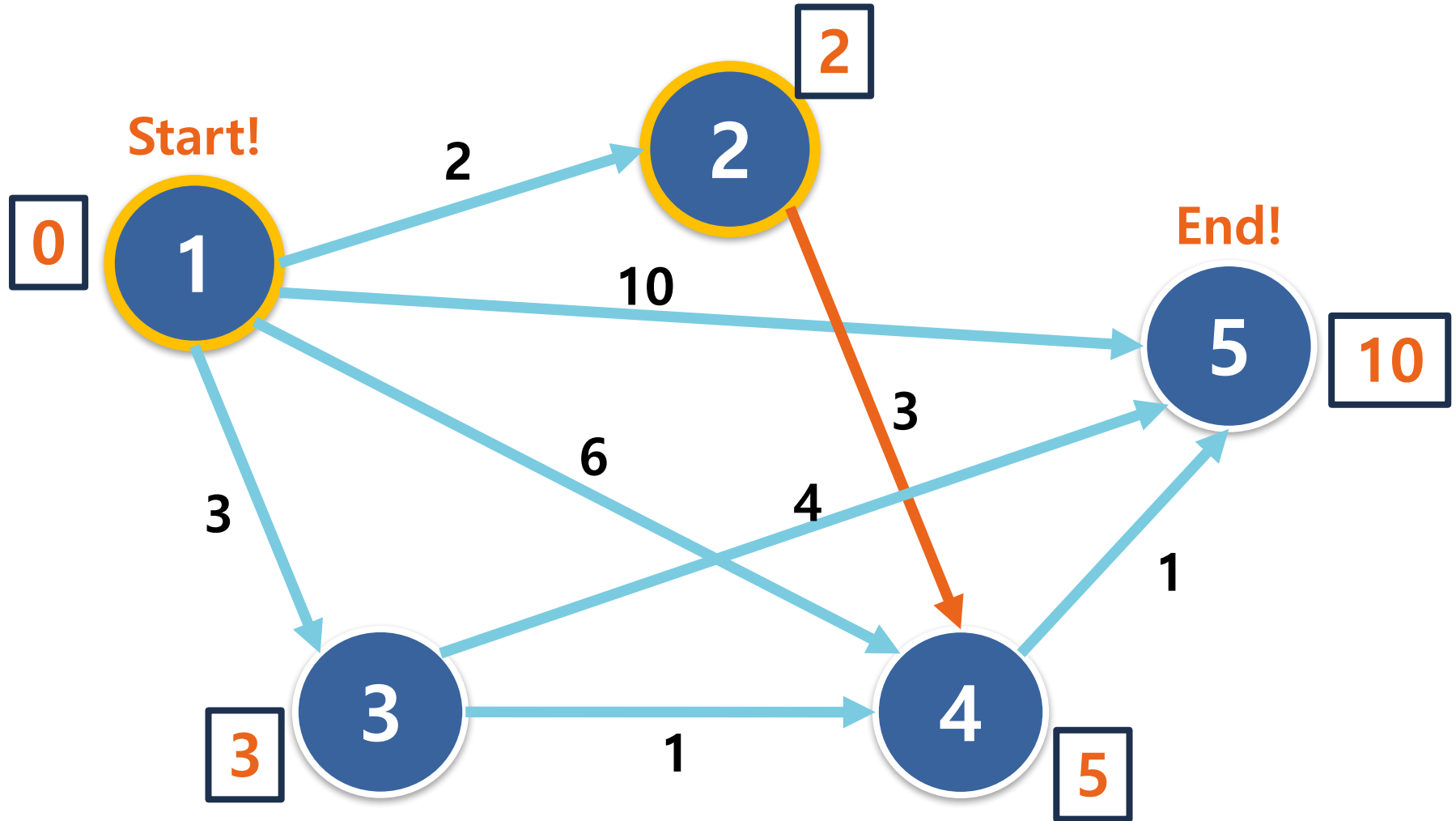
Dijkstra Algorithm - 1번 노드 기준 갱신



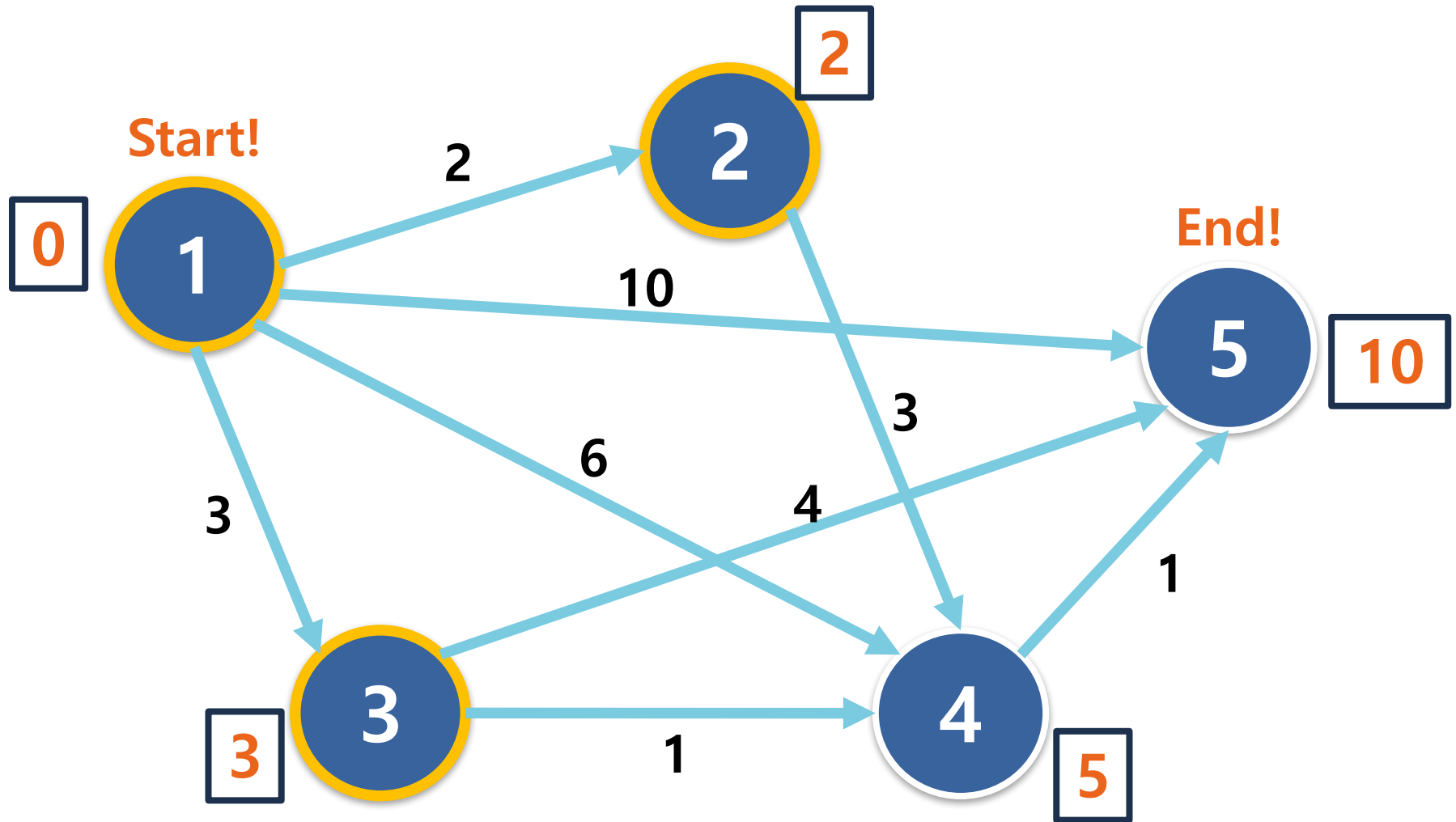
Dijkstra Algorithm - min노드(2) 선택



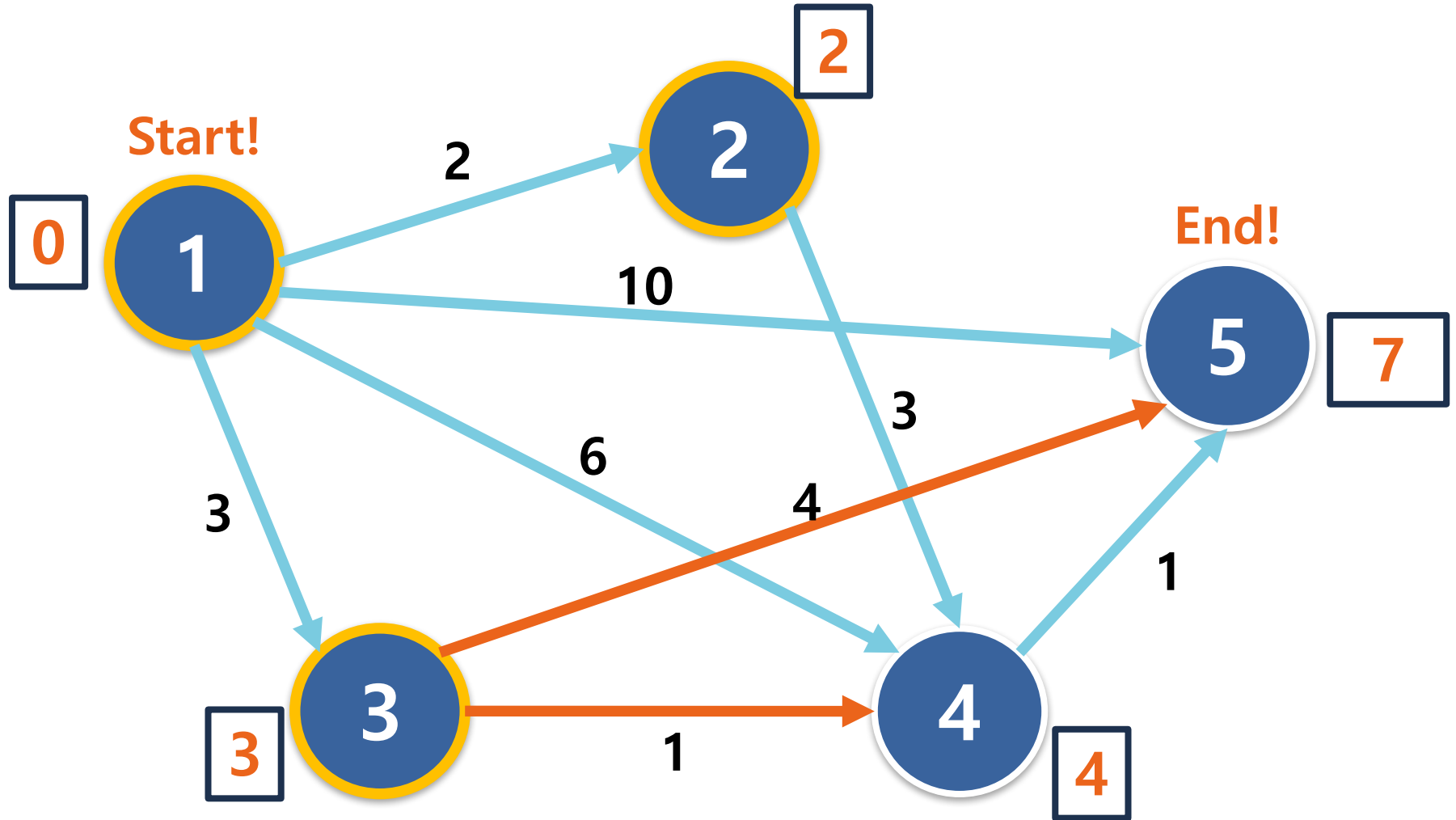
Dijkstra Algorithm - 2번 노드 기준 갱신



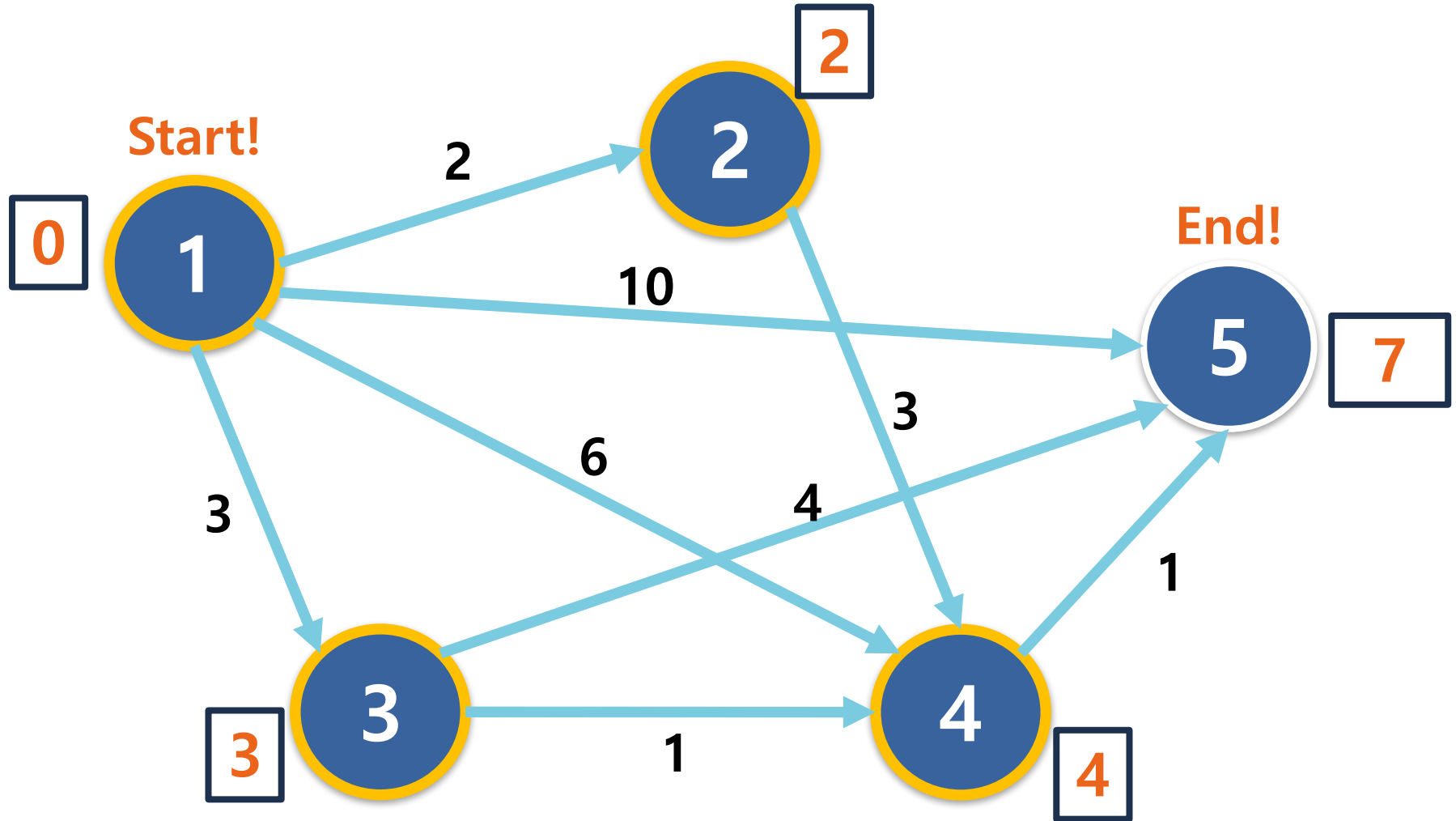
Dijkstra Algorithm - min노드(3) 선택



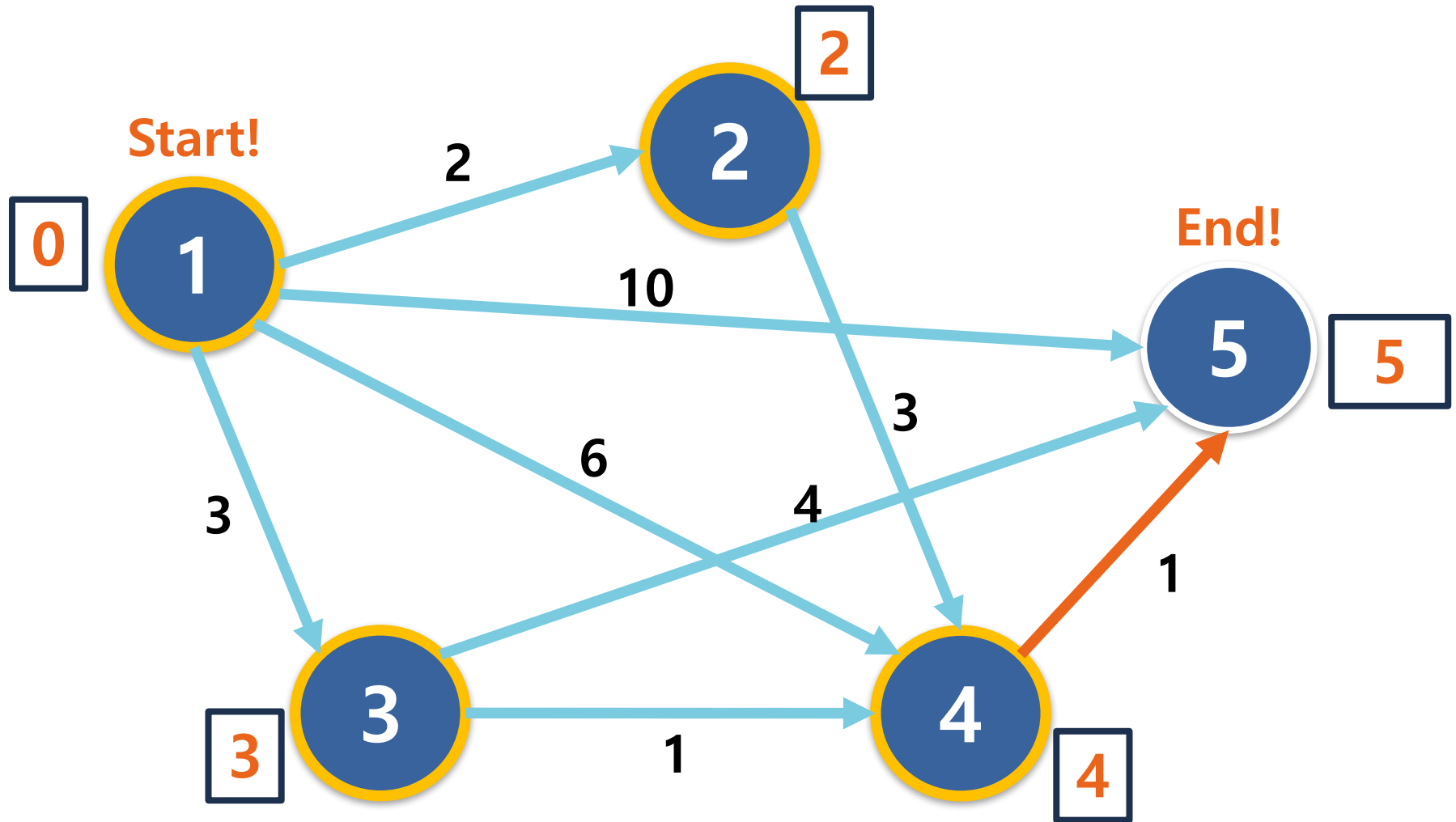
Dijkstra Algorithm - 3번 노드 기준 갱신



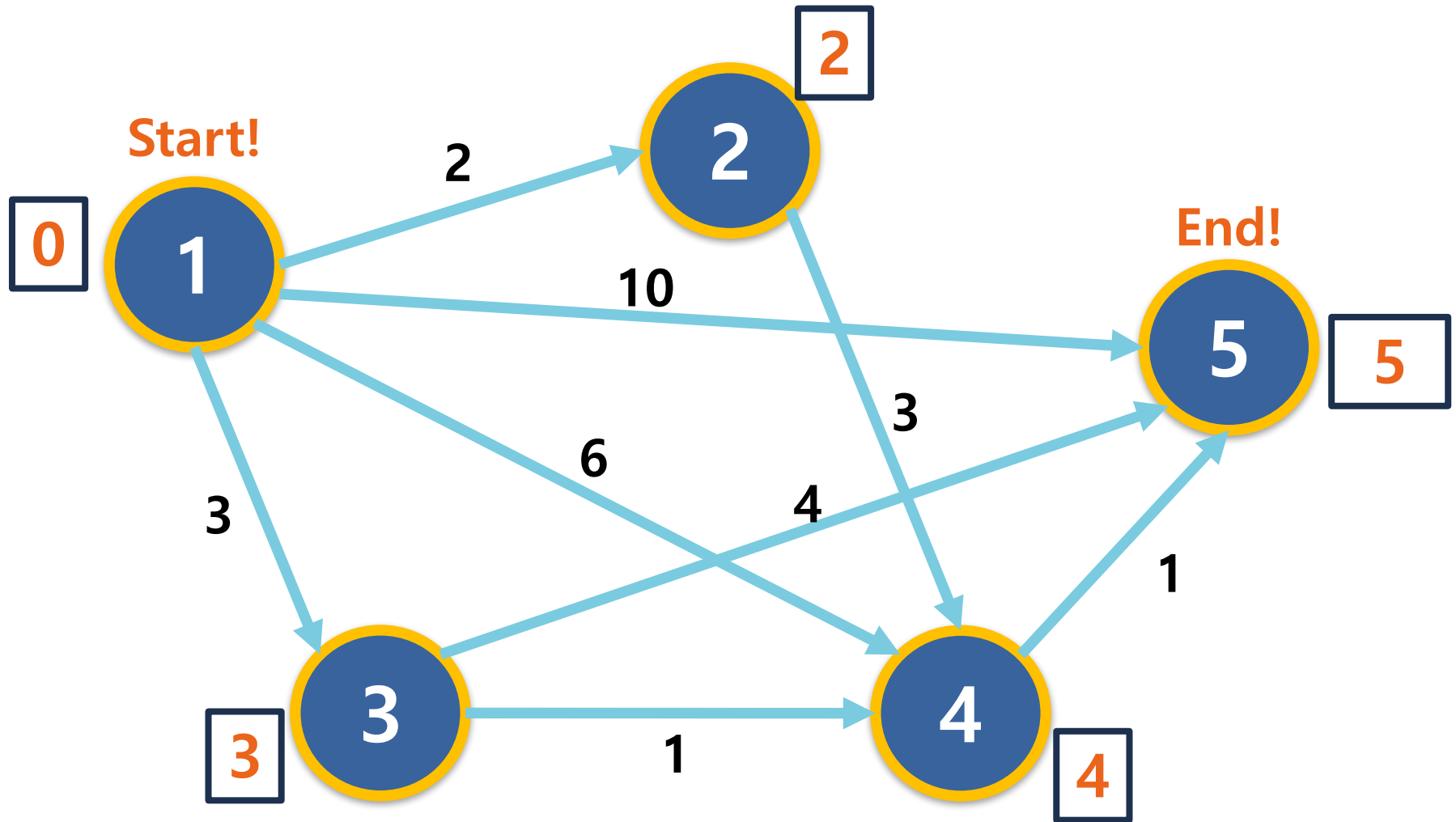
Dijkstra Algorithm - min노드(4) 선택



Dijkstra Algorithm - 4번 노드 기준 갱신



Dijkstra Algorithm - min노드(5) 선택



Dijkstra Algorithm

```
int dijkstra(int start, int end) {
    dist[start] = 0;
    for (int t = 1; t <= n; t++) {
        // Extract Min node
        int min = INF;
        int min_v;
        for (int i = 1; i <= n; i++) {
            if (visit[i] == 0 && min > dist[i]) {
                min = dist[i];
                min_v = i;
            }
        }
        visit[min_v] = 1;

        // Relaxation
        for (int j = 1; j <= n; j++) {
            if (dist[j] > dist[min_v] + node[min_v][j]) {
                dist[j] = dist[min_v] + node[min_v][j];
            }
        }
    }

    return dist[end];
}
```

Assignment 9 - Shortest Path

- 제출방식: Assignment9 폴더 만들고
Assignment9_학번.c 파일 저장
Ex) Assignment9/Assignment9_2016000000.c
- GitLab(<https://hconnect.hanyang.ac.kr/>)으로 제출
- GitLab이 안되면 이메일(roomylee@naver.com) 제출
- 제출기한: 6월 18일 23시 59분

감사합니다
