

Dijkstra Algorithm → 대표적 최단거리 알고리즘

단, degree가
양수!

basic principle of mechanical

1. 출발 도착노드 설정 (example) 노포역 ↔ 부산역
2. '최단 거리 테이블'을 초기화
3. 위치노드에서 방문x 노드 구별, 거리가 가장 짧은 노드 선택
4. 해당노드를 거쳐 다른 노드로 넘어가는 간선 비용(가중치)을 생각해 '최단거리 테이블' 업데이트
5. 3~4 과정 반복

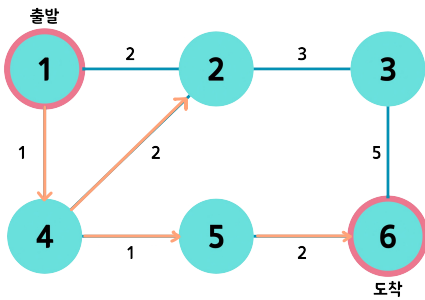


'최단 거리 테이블' → '차운 테이블'
role: 필요한 최단 거리 기록

'노드 방문 여부 체크 배열' → '차운 테이블'
role: 방문여부 True/False

Size
동일

example of mechanical



Node 1

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

기본적으로 거리를 infinity (또는 그에 준하는 매우 큰 수) 에 설정 한다

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

출발노드 거리를 0으로 초기화 시킨다

노드	1	2	3	4	5	6
거리	0	min (inf, 2)	inf	min (inf, 1)	inf	inf
	0	2	inf	1	inf	inf

업데이트

인접한 노드의 거리를 구하고 다음 노드로 넘어간다

Node 4

노드	1	2	3	4	5	6
거리	0	min (2, 1+2)	inf	1	min (inf, 1+1)	inf
	0	2	inf	1	2	inf

업데이트

Node 2와 Node 5 까지 거리 계산

Node 2는 min (2, 3) 여기서, 2는 기존값 3은 2까지 총 거리

Node 5는 min (inf, 2)

if 거리가 동일 시 :

인덱스가 작은 값 (여기서는 node 2)로 간다

Node 2

노드	1	2	3	4	5	6
거리	0	2	min (inf, 2+3)	min (1, 2+2)	2	inf
	0	2	5	1	2	inf

업데이트

Node 4로 가는데 그 다음은

Node 2는 방문처리 되었으니 Node 5이다

Node 5

노드	1	2	3	4	5	6
거리	0	2	5	1	2	min (inf, 2+2)
	0	2	5	1	2	4

업데이트

Result

노드	1	2	3	4	5	6
거리	0	2	5	1	2	4