

1.목표

최단경로 알고리즘과 최소비용 신장트리 알고리즘을 통한 문제 해결

2.과제를 해결하는 방법

- 1) 다익스트라 알고리즘에 대한 이해
- 2) 프림 알고리즘에 대한 이해
- 3) 그래프에 대한 이해

3.과제를 해결한 방법

1) Dijkstra

반복을 통해 dist, prev, qnode를 초기화한다. 그리고 시작 노드의 dist 값을 0으로 설정한다. 모든 노드를 방문할 때까지 반복하며 최소 거리값과 그 값을 가지는 노드를 udist와 unode에 저장한다. 그 후 qnode에서 unode를 제거하고 wdgraph[unode]를 반복하면서, dist[unode]와 wdgraph[unode][nbr] 값이 저장된 alt와 dist를 비교하여 alt가 작다면 dist와 prev에 alt와 unode를 저장한다.

그 후, dist와 prev를 반환한다.

2) Prim

wugraph의 키 값을 반복하며 mst를 초기화하고, mst를 반복하며 remain_node에 node를 저장한다. start_node 값에 랜덤한 값을 저장하고, added_node에 start_node를 추가하고, remain_node에서 start_node를 삭제한다. 그 후 remain_node가 없어질 때까지, 반복을 하며 mst의 값을 설정한다. 마지막으로 mst를 반환한다.

3) Search_Min

src_node,와 dst_node를 None, dist를 float('inf')으로 초기화하고, added_node와 remained_node를 반복하면서 wugraph에서 해당 weight 값과 dist를 비교하여 dist가 더 크다면 dist를 wugraph의 weight 값으로 바꾸고, src_node와 dst_node를 src와 dst로 바꿔서 최소값을 저장한다. 마지막으로 src_node와 dst_node, dist를 반환한다.

4) Dijkstra Main

key1과 key2를 통해 WDGRAPH와 WDGRAPH[key1]를 반복하면서 WDGRAPH의 값을 graph의 엣지 값으로 추가하고, WDGRAPH에 저장된 weight 값을 graph와, data에 저장한다.

list형 변수 path를 만들고, prev에 dijkstra의 두 번째 열값을 저장하여 이전 엣지 값을 저장한다. 그 후, prev를 반복하며 각 엣지의 prev값과 현재 엣지의 값을 path의 저장하여 경로를 나타낸다.

5) Prim Main

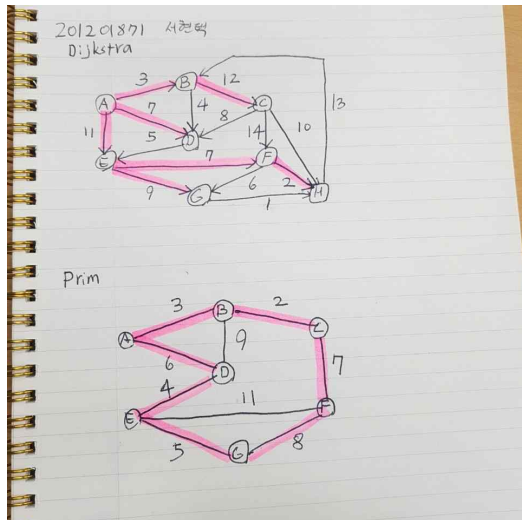
key1과 key2를 통해 WDGRAPH와 WDGRAPH[key1]를 반복하면서 WDGRAPH의 값을

graph의 엣지 값으로 추가하고, WDGRAPH에 저장된 weight 값을 graph와, data에 저장한다.

list형 변수 path를 만들고, prev에 prim값을 저장한다. prev를 이중으로 반복하면서 path의 엣지 값을 저장하여 경로를 나타낸다.

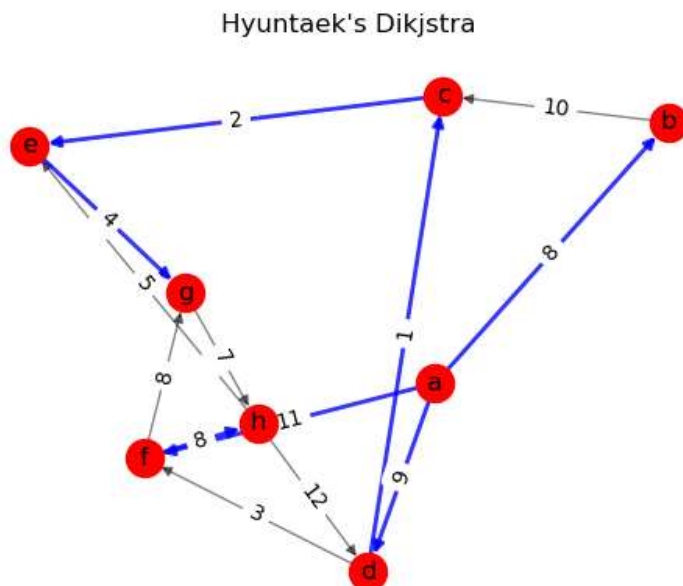
4.결과화면

1) 내가 만든 Dijkstra와 Prim 그래프



2) 제공 Dijkstra 결과, 그래프

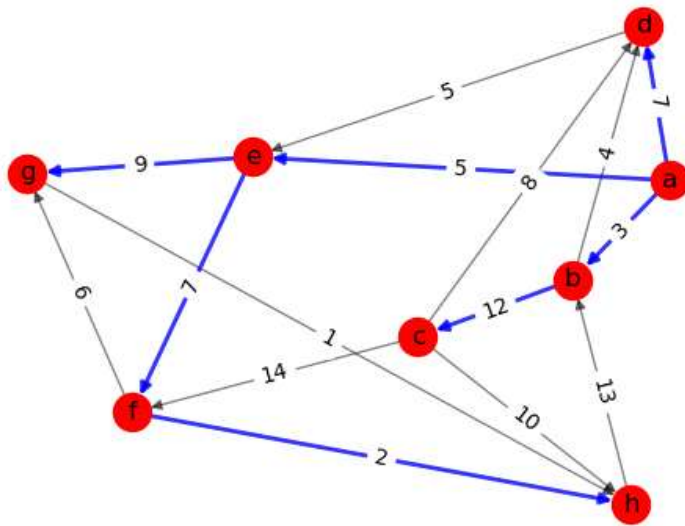
Hyuntaek's Dijkstra
 ({'a': 0, 'b': 8, 'c': 10, 'd': 9, 'e': 12, 'f': 11, 'g': 16, 'h': 19}, {'a': None, 'b': 'a', 'c': 'd', 'd': 'a', 'e': 'c', 'f': 'a', 'g': 'e', 'h': 'f'})



3) 만든 Dijkstra 결과, 그래프

Hyuntaek's Dijkstra
 ({'a': 0, 'b': 3, 'c': 15, 'd': 7, 'e': 5, 'f': 12, 'g': 14, 'h': 14}, {'a': None, 'b': 'a', 'c': 'b', 'd': 'a', 'e': 'a', 'f': 'e', 'g': 'e', 'h': 'f'})

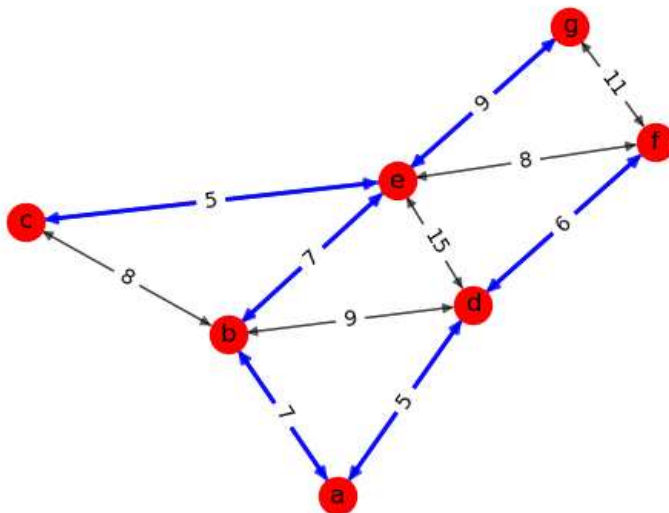
Hyuntaek's Dijkstra



4) 제공 Prim 결과, 그래프

```
Prim 출력
{'a': {'d': 5, 'b': 7}, 'b': {'a': 7, 'e': 7}, 'c': {'e': 5}, 'd': {'a': 5, 'f': 6}, 'e': {'b': 7, 'c': 5, 'g': 9}, 'f': {'d': 6}, 'g': {'e': 9}}
```

Hyuntaek's Prim



5) 만든 Prim 결과, 그래프

```
Prim 출력
{'a': {'b': 3, 'd': 6}, 'b': {'a': 3, 'c': 2}, 'c': {'b': 2, 'f': 7}, 'd': {'a': 6, 'e': 4}, 'e': {'d': 4, 'g': 5}, 'f': {'c': 7}, 'g': {'e': 5}}
```

Hyuntaek's Prim

