**TRP(Traveling Rail-rer Problem) Problem solving**

**문제 정의**

한국의 일반 철도망을 Dynamic graph로 구현하고 이를 활용하여 내일러들에게 보다 효율적인 경로를 제시하고자 함.

**세부 목표**

1. 주어진 목적지들에 대해 최소 가중치를 갖는 경로를 구함

2. 최적 경로와 각 목적지 별로 정해진 체류시간에 대해 제한 시간(3일 또는 5일) 내에 가능한 경로인지 판별

3. 시간이 많이 남을 경우 추가 관광지를 경로상에서 추천

**모델 설정**

각 역을 그래프의 노드로, 역 사이의 연결관계, 즉 철도 노선을 간선으로 한다.

각 간선은 다음과 같은 정보를 저장한다.

노선 번호: 해당 간선이 어느 노선에 포함되어 있는가를 각 노선별로 encoding된 숫자로 표현한다.

영업 거리: 해당 간선에 인접한 노드들 사이의 영업거리

노선의 종류: 단선인지, 복선 이상인지 boolean 형으로 나타낸다.

운행 상태: 정상 운행중인지 Boolean 형으로 나타낸다.

**문제 해결을 위한 가정**

1. 각 구간의 가중치는 기본적으로 국토교통부 고시의 철도거리표 상의 거리에 따른다.

2. 단선인 구간의 가중치는 복선인 구간의 가중치의 2.1배로 가정한다.

2-1. 복복선 이상의 구간은 복선과 동일하게 취급한다.

3. 운휴중인 역이 포함된 구간의 가중치는 INF로 둔다.

4. 폐선이 되거나 선로 개량의 결과 폐역 된 역이 포함된 구간은 노드 삭제를 통해 반영한다.

5. 한 역을 두 번 이상 체류하지 않는다. 즉 목적지로 설정된 역들은 모두 서로 다르다.

5-1. 단 경유나 환승은 가능하다.

6. 무향 그래프를 통해 나타낸다.

7. 체류 시간이 지정되지 않았다면 4시간을 기본으로 가정한다.

8. 영동선, 태백선 등 산악을 지나는 구간이 대부분인 노선의 가중치를 같은 길이의 경로보다 1.1배로 설정한다.

9. 목적지로 지정된 역 중 첫 역을 출발지로 가정하고, 출발지와 도착지는 같다고 가정한다.

**문제 해결 과정**

0. 기본적인 문제 해결 과정은 다익스트라 알고리즘과 해밀턴 경로를 활용한다.

1. 모든 역을 순서대로 1, 2, 3, …, n이라고 번호를 붙인다. 인 두 역 i, j에 대해 두 역 사이의 거리를 나타내는 배열 d[i][j]를 만든다.

2. 인 두 역 i, j에 대해 다음과 같은 과정을 반복한다.

a. 만약 두 역 i, j가 같은 노선에 접속해 있다면, 철도거리표에서 두 역 간의 거리를 가져와 d[i][j]를 초기화 한다.

b. 그렇지 않다면, d[i][j]=INF로 초기화한다.

3. 인 두 역 i, j에 대해 다익스트라 알고리즘을 반복한다.

4. 목적지로 지정된 역들을 노드로, 과정 3에서 구한 그 역 사이의 거리를 간선의 가중치로 갖는 새 그래프를 그린다.

5. 4에서 그린 그래프에 대해, 해밀턴 경로를 그린다.