

Group Activity 11

(3인 혹은 4인으로 팀을 구성하여 아래의 문제를 푼다. 팀 구성은 매 시간마다 달라져도 된다.)

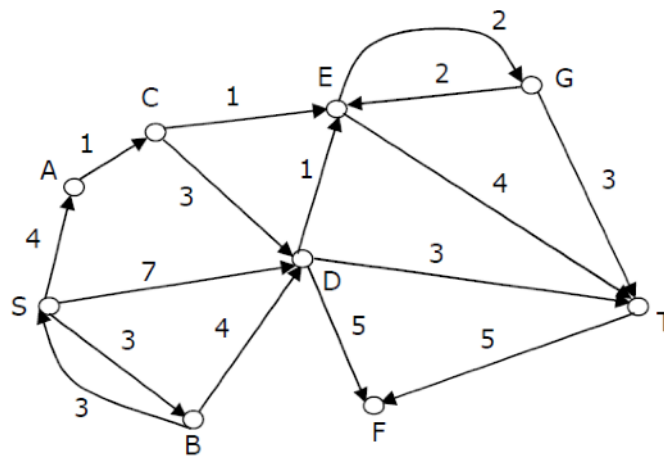
팀원1: _____

팀원2: _____

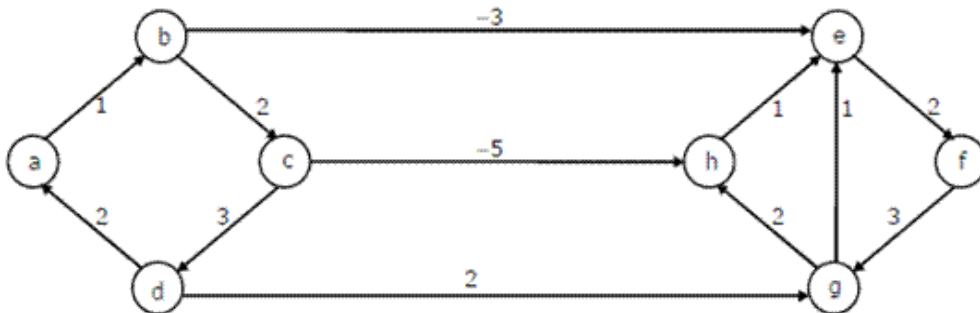
팀원3: _____

팀원4: _____

- 다음의 그래프에서 S 에서 T 까지 가는 여러 개의 최단경로가 존재한다. Dijkstra의 알고리즘으로 S 에서 T 까지 가는 최단경로를 찾는다면 어떤 경로가 찾아지는가? 단, Dijkstra의 알고리즘에서 더 짧은 경로가 찾아질 때만 relaxation 연산을 적용한다고 가정한다. 즉 $d[v] > d[u] + w(u, v)$ 일 때만 $d[v]$ 를 $d[u] + w(u, v)$ 로 갱신한다 (즉, $d[v] = d[u] + w(u, v)$ 일 때는 $d[v]$ 를 갱신하지 않는다.)



- 음수 가중치가 있는 다음의 그래프의 정점 a 에서 Dijkstra의 알고리즘을 시작한다면 a 로 부터의 최단 경로가 올바르게 계산되어지는 정점은 어떤 것들인가?



3. 최단 경로에 대해서 다음 중 옳은 것을 모두 고르면? 이유는?

(a) 모든 에지의 가중치를 각각 1만큼 증가 시켜도 정점들 간의 최단 경로는 바뀌지 않는다.

(b) 모든 에지의 가중치에 2를 곱해도 정점들 간의 최단 경로는 바뀌지 않는다.

4. 일반적인 그래프에서 최장경로(longest path)를 구하는 것은 어려운 문제이다. 하지만 DAG에 최장 경로를 구하는 것은 비교적 간단하다. 다음은 DAG에서 임의의 정점 s 로 부터 다른 모든 정점으로의 최장 경로를 구하는 알고리즘이다.

1) 먼저 모든 정점들에 대해서 다음과 같이 dist 값을 초기화한다.

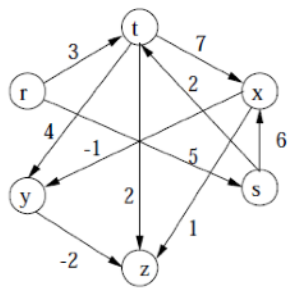
$$\text{dist}[v] = \begin{cases} 0 & \text{if } v = s; \\ -\infty & \text{otherwise.} \end{cases}$$

2) 정점들의 위상순서를 구한다.

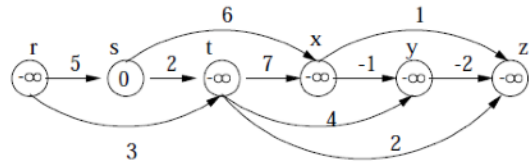
3) 위상순서에 따라서 순서대로 정점들을 고려한다. 정점 u 에 대해서 다음과 같이 실행한다.

```
for each vertex v adjacent from u do
    if dist[v] < dist[u] + weight(u, v)
        dist[v] = dist[u] + weight(u, v)
```

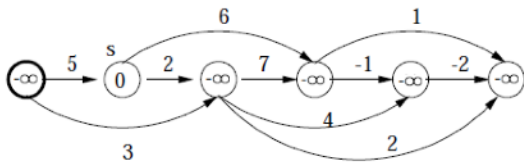
아래의 그림은 이 알고리즘의 실행 예이다. 이 알고리즘이 왜 최장경로를 올바르게 구하는지 설명하라.



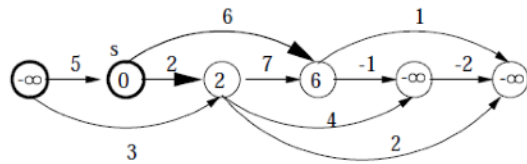
(a)



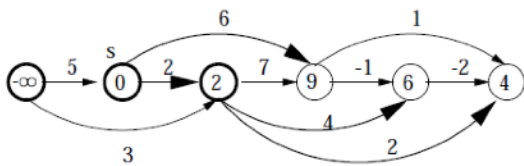
(b)



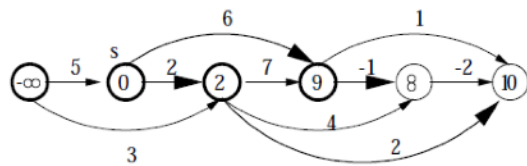
(c)



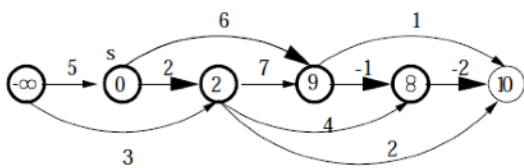
(d)



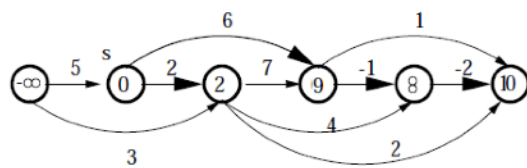
(e)



(f)



(g)



(h)