



03. Shortest Path

2019 Summer / 20141574 임지환



#10217 KCM Travel

- <https://www.acmicpc.net/problem/10217>
- 공항의 수 N ($2 \leq N \leq 100$)
- 총 지원비용 M ($0 \leq M \leq 10,000$)



#10217 KCM Travel

비용 문제가 없다면?

⇒ 단순 최단경로 문제.

```
if (dist[n] == INF) cout << "Poor KCM\n";  
else cout << dist[n] << '\n';
```



#10217 KCM Travel

$\text{dist}[\text{pos}][\text{currency}]$:= pos번 공항에 currency만큼의 비용을 갖고 도착하였을 때의 최단 이동거리

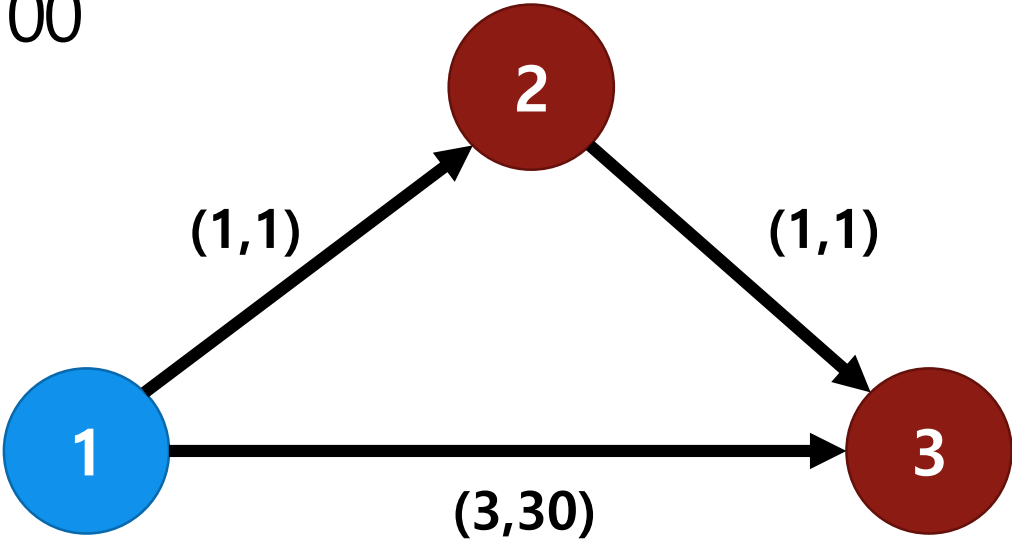
initial state : $\text{dist}[1][M]$

destination : $\text{dist}[n][0] \sim \text{dist}[n][M]$



#10217 KCM Travel

Currency : 100



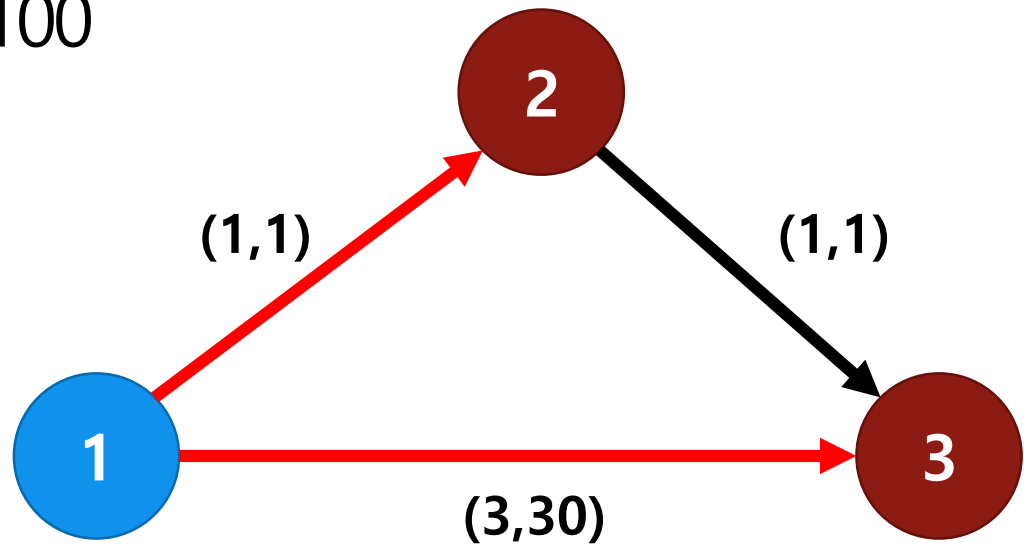
(pos, currency)	distance
(1,100)	0

(dist[pos][currency], pos, currency)	(0, 1, 100)				
--------------------------------------	-------------	--	--	--	--



#10217 KCM Travel

Currency : 100



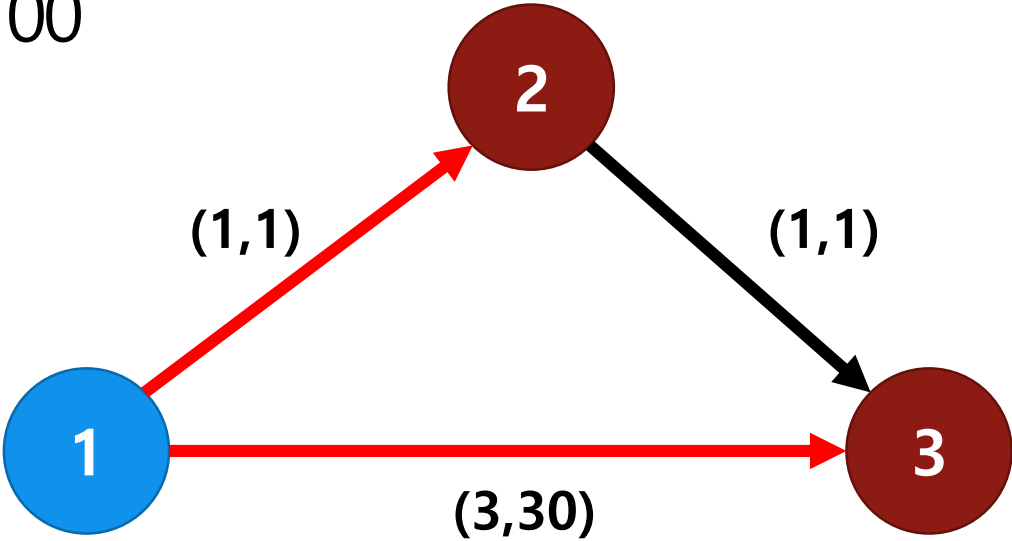
(pos, currency)	distance
(1,100)	0

(dist[pos][currency], pos, currency)



#10217 KCM Travel

Currency : 100



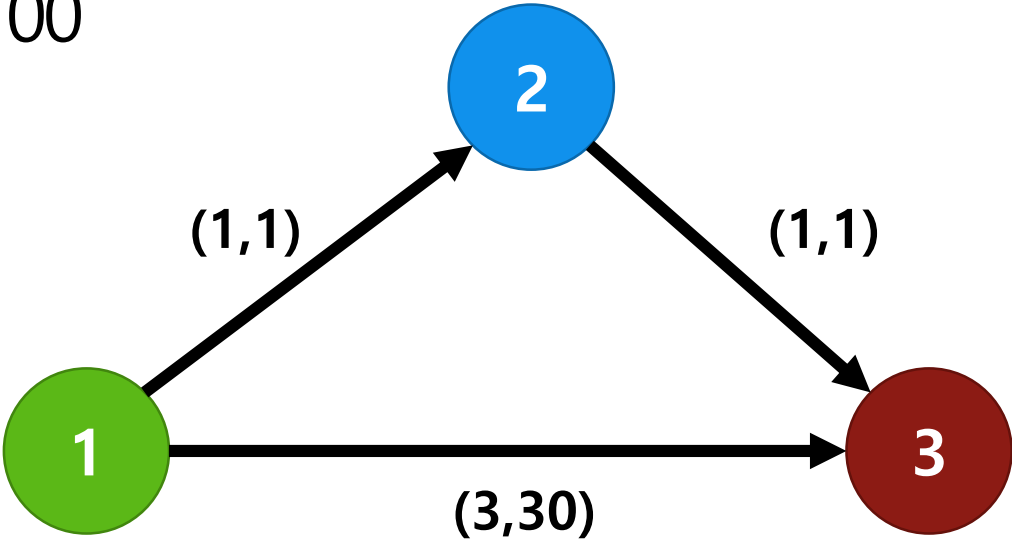
(pos, currency)	distance
(1,100)	0
(2,99)	1
(3,97)	30

(dist[pos][currency], pos, currency)	(1, 2, 99)	(30, 3, 97)			
--------------------------------------	------------	-------------	--	--	--



#10217 KCM Travel

Currency : 100



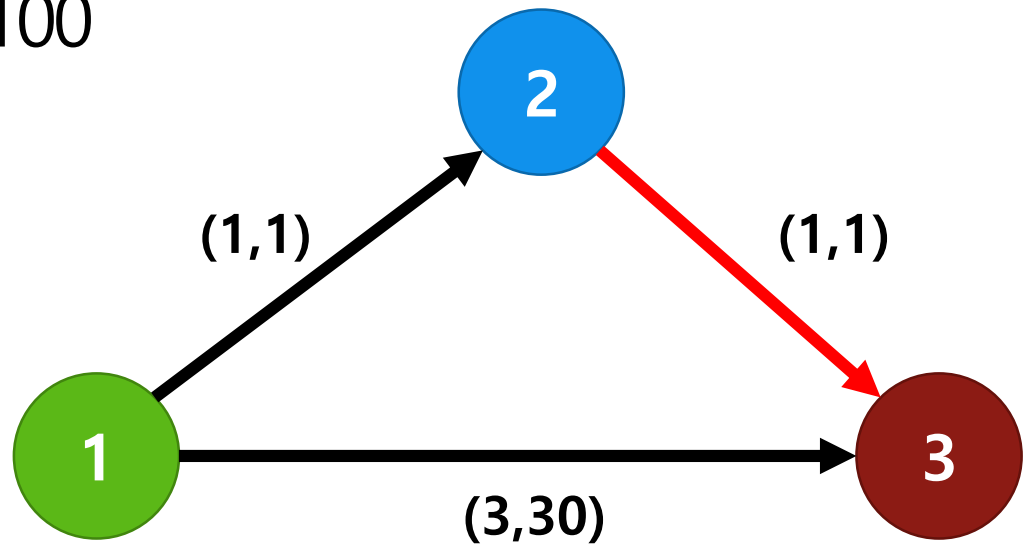
(pos, currency)	distance
(1,100)	0
(2,99)	1
(3,97)	30

(dist[pos][currency], pos, currency)	(1, 2, 99)	(30, 3, 97)			
--------------------------------------	------------	-------------	--	--	--



#10217 KCM Travel

Currency : 100



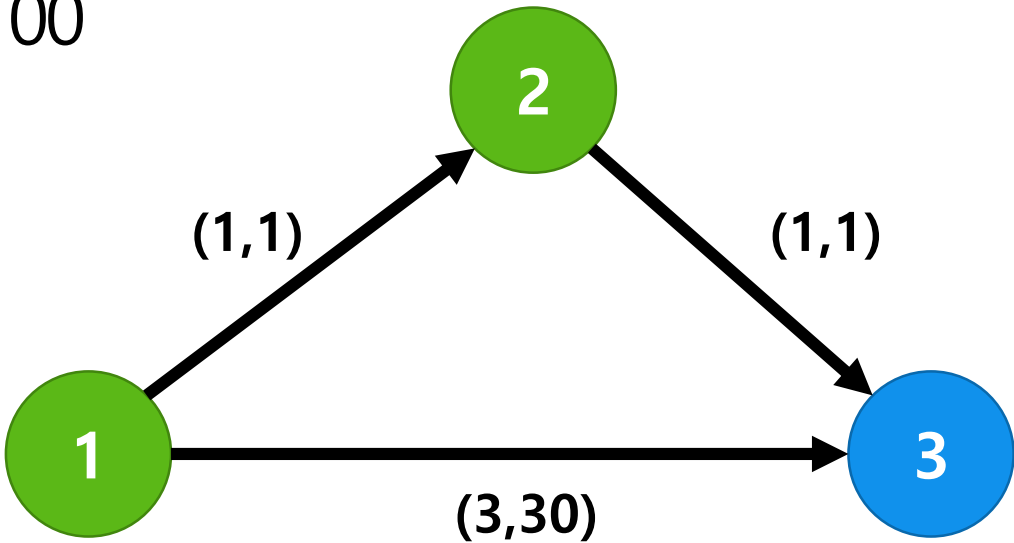
(pos, currency)	distance
(1,100)	0
(2,99)	1
(3,97)	30

(dist[pos][currency], pos, currency)	(30, 3, 97)				
--------------------------------------	-------------	--	--	--	--



#10217 KCM Travel

Currency : 100



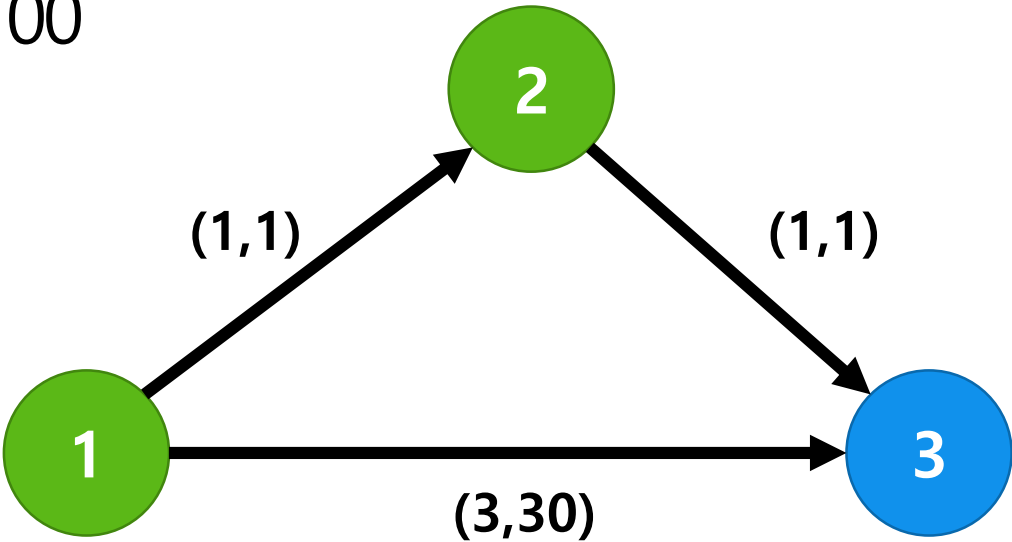
(pos, currency)	distance
(1,100)	0
(2,99)	1
(3,98)	30
(3,97)	2

(dist[pos][currency], pos, currency)	(2, 3, 98)	(30, 3, 97)			
--------------------------------------	------------	-------------	--	--	--



#10217 KCM Travel

Currency : 100



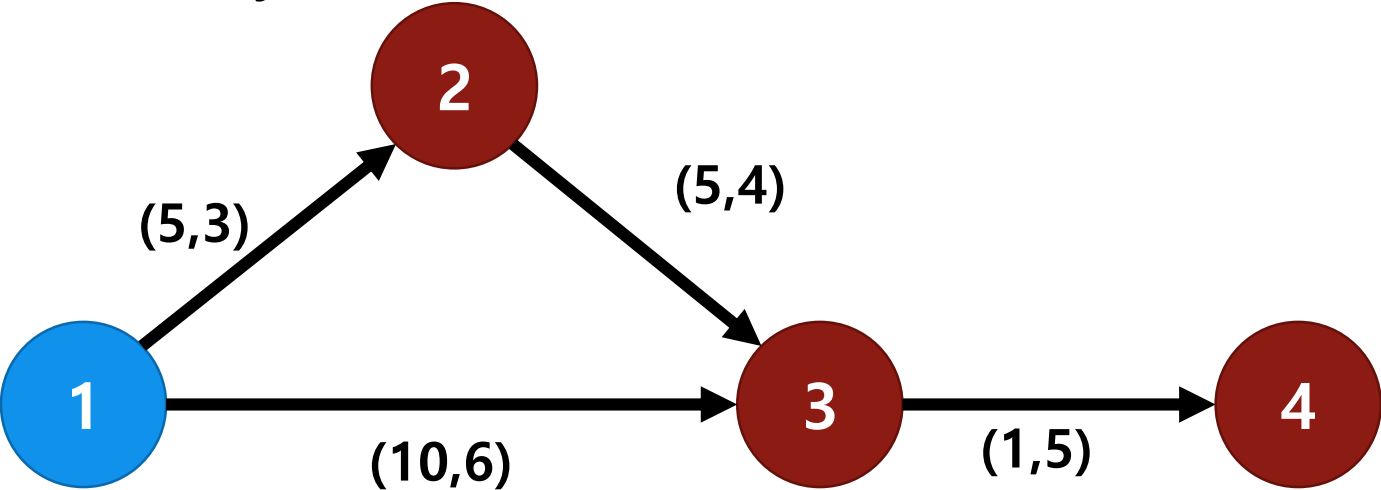
(pos, currency)	distance
(1,100)	0
(2,99)	1
(3,98)	30
(3,97)	2

(dist[pos][currency], pos, currency)	(2, 3, 98)	(30, 3, 97)			
--------------------------------------	------------	-------------	--	--	--



#10217 KCM Travel

Currency : 10



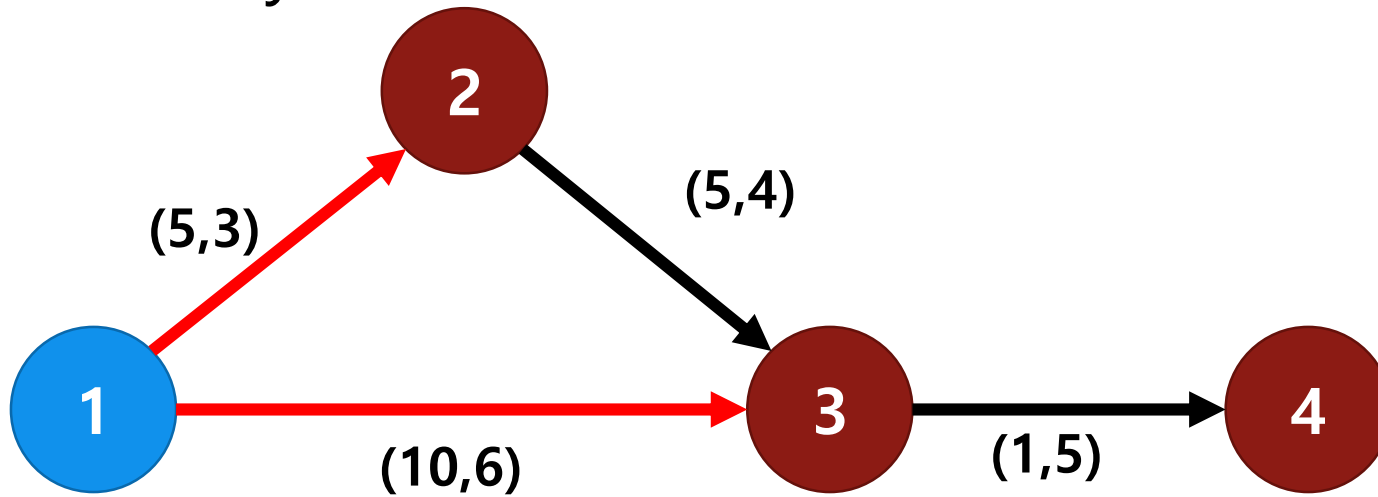
(pos, currency)	distance
(1,10)	0

(dist[pos][currency], pos, currency)	(0, 1, 10)				
--------------------------------------	------------	--	--	--	--



#10217 KCM Travel

Currency : 10



(pos, currency)	distance
(1,10)	0

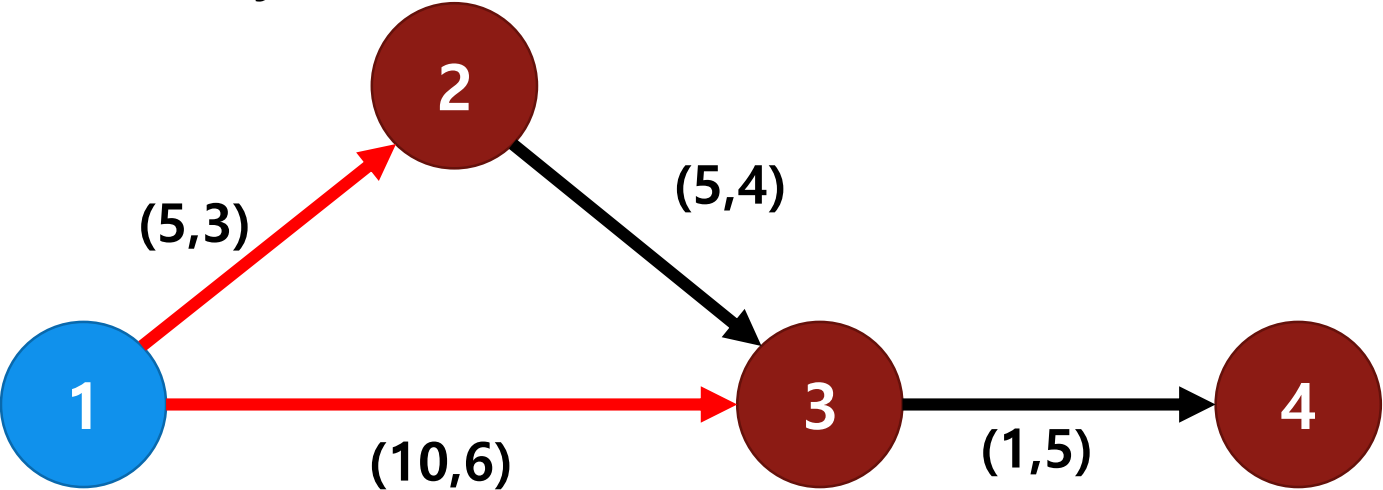
(dist[pos][currency], pos, currency)

--	--	--	--	--



#10217 KCM Travel

Currency : 10



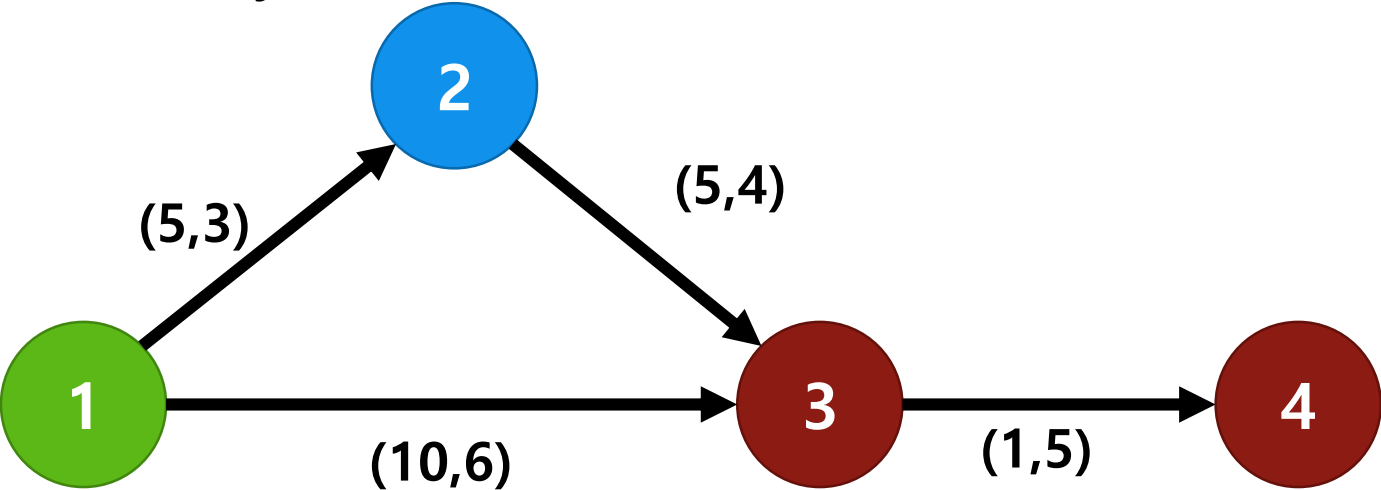
(pos, currency)	distance
(1,10)	0
(2,5)	3
(3,0)	6

(dist[pos][currency], pos, currency)	(3, 2, 5)	(6, 3, 0)			
--------------------------------------	-----------	-----------	--	--	--



#10217 KCM Travel

Currency : 10



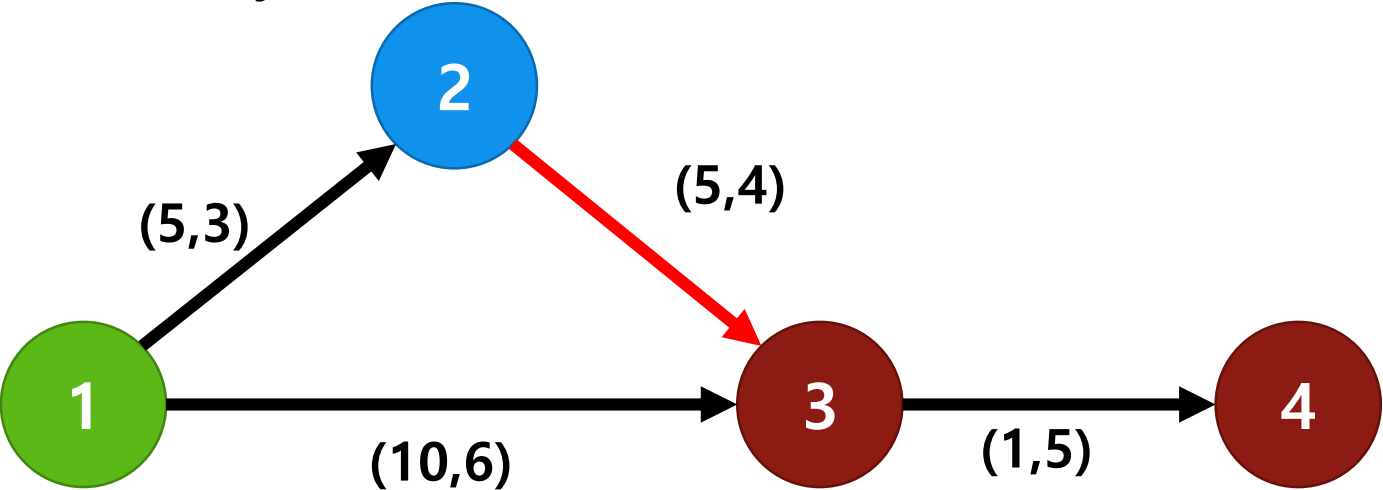
(pos, currency)	distance
(1,10)	0
(2,5)	3
(3,0)	6

(dist[pos][currency], pos, currency)	(3, 2, 5)	(6, 3, 0)			
--------------------------------------	-----------	-----------	--	--	--



#10217 KCM Travel

Currency : 10



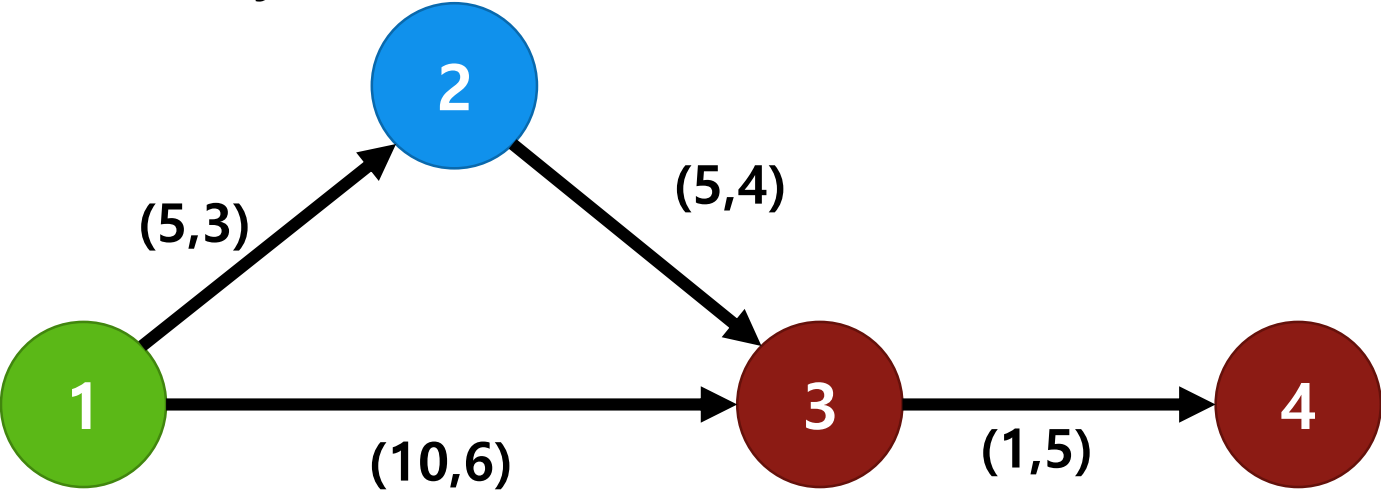
(pos, currency)	distance
(1,10)	0
(2,5)	3
(3,0)	6

(dist[pos][currency], pos, currency)	(6, 3, 0)				
--------------------------------------	-----------	--	--	--	--



#10217 KCM Travel

Currency : 10



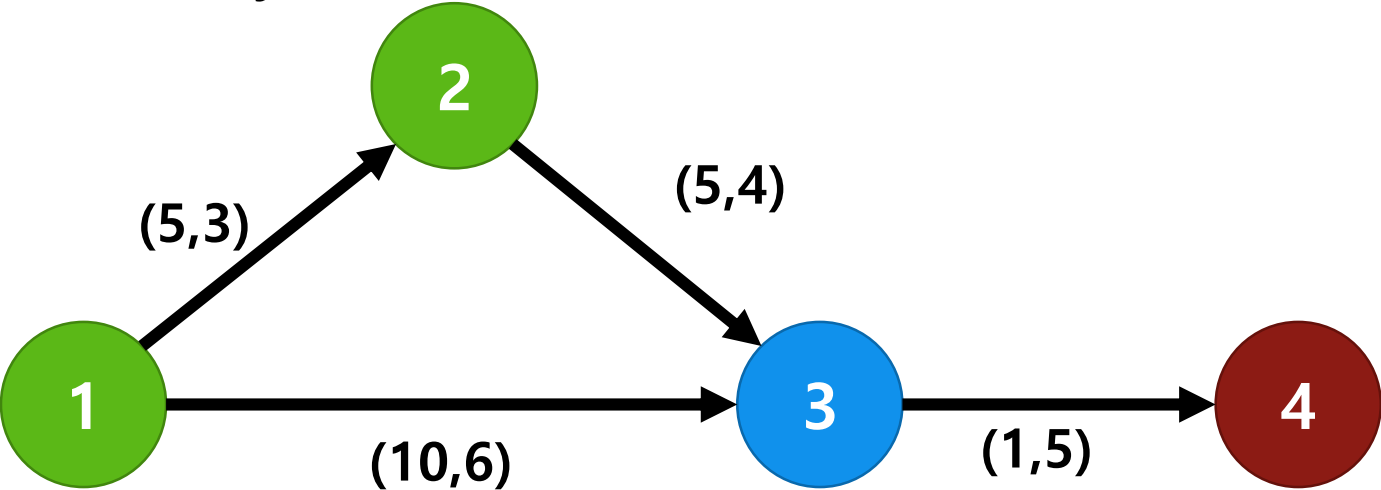
(pos, currency)	distance
(1,10)	0
(2,5)	3
(3,0)	6

(dist[pos][currency], pos, currency)	(6, 3, 0)				
--------------------------------------	-----------	--	--	--	--



#10217 KCM Travel

Currency : 10



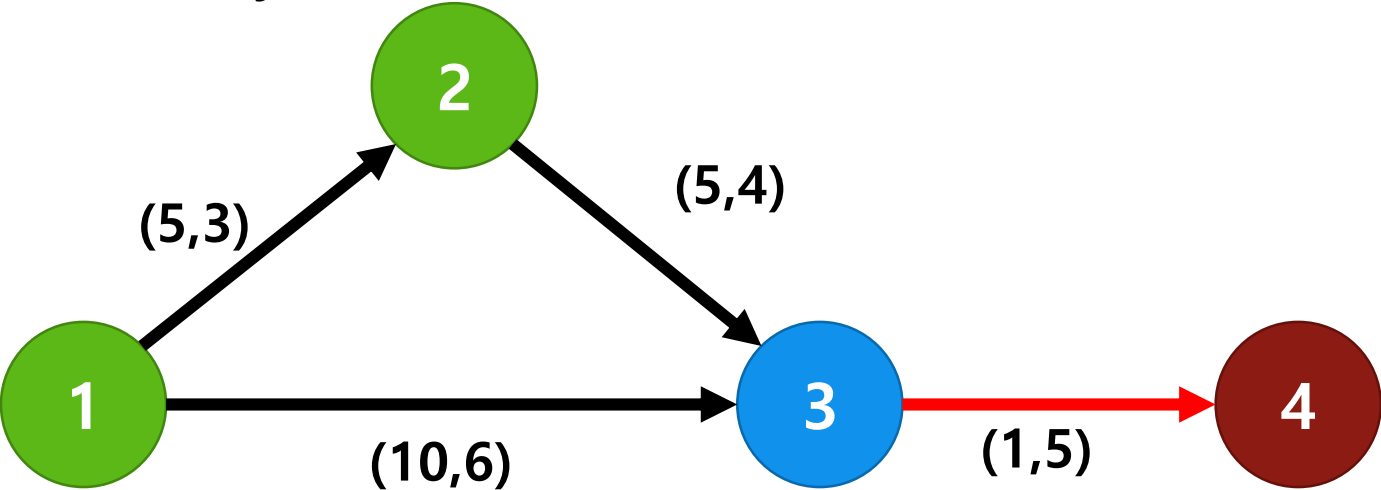
(pos, currency)	distance
(1,10)	0
(2,5)	3
(3,0)	6

(dist[pos][currency], pos, currency)	(6, 3, 0)				
--------------------------------------	-----------	--	--	--	--



#10217 KCM Travel

Currency : 10



(pos, currency)	distance
(1,10)	0
(2,5)	3
(3,0)	6

(dist[pos][currency], pos, currency)     



#10217 KCM Travel

현재 pos번 위치에서 currency만큼의 돈을 갖고 있을 때
연결된 다른 위치로 가는 비용보다 currency가 크다면 이동 가능

```
if (currency - cost < 0) continue;
if (dist[next][currency - cost] > dist[cur][currency] + d) {
    dist[next][currency - cost] = dist[cur][currency] + d;
    pq.push(III(dist[next][currency - cost],
                ii(next, currency - cost)));
}
```



Sogang **ACM-ICPC** Team

#10217 KCM Travel

<http://boj.kr/b9ccb3cfa3234b01834f58ac8b2e5ad1>



#6598 Arbitrage

Arbitrage란 '차익 거래' 를 의미한다.

예를 들어 1 USD로 0.5 British pound를 살 수 있고, 1 British pound로 10.0 French franc를 살 수 있고, 1 French franc로 0.21 USD를 살 수 있다면

1 USD로부터 시작하여 $1 * 0.5 * 10.0 * 0.21$ 의 과정을 거쳐 다시 1 USD를 구매하였을 때의 가치는 1.05로 5%의 이익을 볼 수 있다.

화폐 간 환율이 주어졌을 때 arbitrage가 존재하는지 구하여라. ($1 \leq N \leq 30$)

원문 : <https://www.informatik.uni-ulm.de/acm/Locals/1996/arbitrage.html>



#6598 Arbitrage

모든 화폐 관계에 대해서 환율을 적용한 후 자기 자신으로 돌아왔을 때의 사이클을 C 라 한다면

$$\prod_{e \in C} w(e) > 1$$

을 만족하는 화폐가 있는지 확인.



#6598 Arbitrage

N 이 30 이하라 모든 환율 이동관계를 전부 고려해도 시간초과X, but



#6598 Arbitrage

N 이 30 이하라 모든 환율 이동관계를 전부 고려해도 시간초과X, but

- 1) 경로 이동 과정이 덧셈이 아니라 곱셈
- 2) 경로가 작아지는 것이 아니라 커지는 것을 찾아야 함



#6598 Arbitrage

$$\prod_{e \in C} w(e) > 1$$



#6598 Arbitrage



$$\prod_{e \in C} w(e) > 1$$

양변에 로그



#6598 Arbitrage

$$\prod_{e \in C} w(e) > 1 \quad \Rightarrow \quad \sum_{e \in C} \log w(e) > 0$$



#6598 Arbitrage

$$\prod_{e \in C} w(e) > 1 \quad \Rightarrow \quad \sum_{e \in C} \log w(e) > 0$$

양변에 음수



#6598 Arbitrage

$$\prod_{e \in C} w(e) > 1 \quad \Rightarrow \quad \sum_{e \in C} \log w(e) > 0 \quad \Rightarrow \quad -\sum_{e \in C} \log w(e) < 0$$



#6598 Arbitrage

- 간선 정보 저장 : 환율에 로그를 취한 값의 음수
- 플로이드를 통한 최단경로 갱신 후
dist[i][i] 값이 음수인 경로 확인



#6598 Arbitrage

<https://raararaara.blog.me/221505865058>



Problem Set

4 11779 최소비용 구하기 2

1 10853 Change of Scenery

1 1854 K번째 최단경로 찾기

4 10849 A Journey to Greece

1 10217 KCM Travel

2 6598 Arbitrage

2 1602 도망자 원숭이

5 13141 Ignition

3 1219 오민식의 고민

5 13907 세금

? 3860 할로윈 묘지



Last week's Solution

5 [11723 집합](#)

2 [15787 기차가 어둠을 헤치고 은하수를](#)

4 [10275 Gold Rush](#)

2 [1194 달이 차오른다, 가자.](#)

1 [2098 외판원 순회](#)

5 [1562 계단수](#)

1 [1102 발전소](#)

? [10464 XOR](#)

5 [11191 XOR Maximization](#)

4 [16685 XOR 포커](#)

5 [13505 두 수 XOR](#)

3 [12844 XOR](#)



Hints for Given Problem

다음 장에는 **스포일러**를 포함하고 있습니다.



#11779 최소비용 구하기 2



- tracking을 하려면 필연적으로 역간선(reverse edge)가 필요합니다.



#10853 Change of Scenery

- 최단경로일 때의 이동거리가 두번 이상 증가하는지 체크해야 합니다.
- 방문여부 체크를 boolean으로 하지 않고 int로 해줍니다.



#1854 K번째 최단경로 찾기



- 앞의 문제와 거의 똑같습니다.



#10849 A Journey to Greece



- 방문하고자 하는 지점수가 최대 16이다.
- 코드를 무작정 베끼라고는 안했습니다.

<https://raararaara.blog.me/221538648815>



#1602 도망자 원숭이

- "원숭이가 도망가는 경로 상에 있는 모든 도시들 중에서 가장 오랜 시간 동안 괴롭힐 수 있는 도시에서 괴롭히기로 계획했다."
- 도시의 개수 N ($2 \leq N \leq 500$)



#13141 Ignition



- 1학기 고급 3주차 강의자료 참고.



#13907 세금



- 통행료에 따른 경로합 구하기와 세금 인상을 함께 할 필요가 있는가?



#3860 할로윈 묘지

- 풀고 저도 좀 알려주시면 감사드리겠습니다.