

Lab 13 Shortest Path

1. 데이터는 다음 데이터 사용

COST MATRIX 1

	0	1	2	3	4	5
0	Max	20	15	Max	Max	Max
1	Max	Max	Max	Max	10	30
2	Max	4	Max	Max	Max	10
3	Max	Max	Max	Max	Max	Max
4	Max	Max	Max	15	Max	Max
5	Max	Max	Max	4	10	Max

COST MATRIX 2

	0	1	2	3	4	5
0	Max	50	10	Max	45	Max
1	Max	Max	15	Max	10	Max
2	20	Max	Max	15	Max	Max
3	Max	20	Max	Max	35	Max
4	Max	Max	Max	30	Max	Max
5	Max	Max	Max	3	Max	Max

2. 기본 알고리즘

1) main()

- print adjacency matrix of the Graph
- call **shortest** (starting vertex '0') //시작 정점은 0

2) Shortest('0')

- 초기화 (visited[7] <= false, dist[0] <= cost[0][I])
- while (< maxN) // maxN = 전체 노드 수
 - . u <= **choose** () //find min node
 - . mark 'u' in the visited[];
 - . for (i= 0 ~ maxN) find shortest distance //노트 algorithm 참조
 - . print dist[i]

3) choose()

```
for (all vertex)
    if not visited, find min node
return min node
```

3. Output 은 다음과 같다.

1) Adjacency Matrix => Cost matrix1 출력

Shortest path distance from Graph 1

Select Node : 2,	Distance ->	0	19	15	999	999	25
Select Node : 1,	Distance ->	0	19	15	999	29	25
Select Node : 5,	Distance ->	0	19	15	29	29	25
Select Node : 3,	Distance ->	0	19	15	29	29	25
Select Node : 4,	Distance ->	0	19	15	29	29	25

2) Adjacency Matrix => Cost matrix2 출력

Shortest path distance from Graph 2

start from vertex 0 :	Distance ->	0	50	10	999	45	999
Select Node : 2	Distance ->	0	50	10	25	45	999
Select Node : 3	Distance ->	0	45	10	25	45	999
Select Node : 4	Distance ->	0	45	10	25	45	999
Select Node : 1	Distance ->	0	45	10	25	45	999
Select Node : 5	Distance ->	0	45	10	25	45	999

4. Screen Shot

```
***** Adjacency matrix *****
999 20 15 999 999 999
999 999 999 999 10 30
999 4 999 999 999 10
999 999 999 999 999 999
999 999 999 15 999 999
999 999 999 4 10 999

***** Shortest Path *****

start from vertex 0 : Distance -> 0 20 15 999 999 999
Select Node : 2 Distance -> 0 19 15 999 999 25
Select Node : 1 Distance -> 0 19 15 999 29 25
Select Node : 5 Distance -> 0 19 15 29 29 25
Select Node : 3 Distance -> 0 19 15 29 29 25
Select Node : 4 Distance -> 0 19 15 29 29 25
```

***** Adjacency matrix *****

```
999 50 10 999 45 999
999 999 15 999 10 999
20 999 999 15 999 999
999 20 999 999 35 999
999 999 999 30 999 999
999 999 999 3 999 999
```

***** Shortest Path *****

start from vertex 0 : Distance -> 0 50 10 999 45 999

Select Node : 2 Distance -> 0 50 10 25 45 999

Select Node : 3 Distance -> 0 45 10 25 45 999

Select Node : 4 Distance -> 0 45 10 25 45 999

Select Node : 1 Distance -> 0 45 10 25 45 999

Select Node : 5 Distance -> 0 45 10 25 45 999

0: 999 50 10 999 45 999 1: 999 999 15 999 10 999 2: 20 999 999 15 999 999 3: 999 20 999 999 35 999 4: 999 999 999 30 999 999 5: 999 999 999 3 999 999