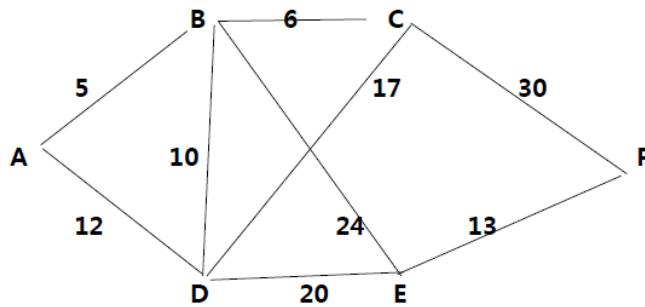


## Lab#11 Prim's Algorithm (Minimal Spanning Tree)

1. Input data 는 다음 그래프를 사용할 것



2. Information for the Prim's Algorithm

1) main()

- print weighted Graph()

2) call prim(A) // Starting vertex A

// 알고리즘 강의노트 참조

### 3. Output 은 다음과 같다.

- 1) Weighted Graph 를 다음과 같이 출력할 것

|   | A   | B   | C   | D   | E   | F   |
|---|-----|-----|-----|-----|-----|-----|
| A | 0   | 5   | 100 | 12  | 100 | 100 |
| B | 5   | 0   | 6   | 10  | 24  | 100 |
| C | 100 | 6   | 0   | 17  | 100 | 30  |
| D | 12  | 10  | 17  | 0   | 20  | 100 |
| E | 100 | 24  | 100 | 20  | 0   | 13  |
| F | 100 | 100 | 30  | 100 | 13  | 0   |

- 2) Minimal Spanning Tree (A 에서 시작, Greedy algorithm)

MST: (A,B) → (B,C) → (C,D) → (D,E) → (E,F)

Weight: 61

#### ● Extra Points (10points 추가):

- Advanced Prim's Algorithm (A 에서 시작, 모든 후보군 함께 고려함)

MST: (A,B) → (B,C) → (A,D) → (D,E) → (E,F)

Weight: 56