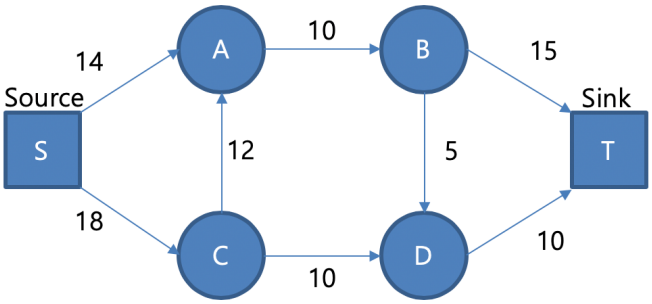
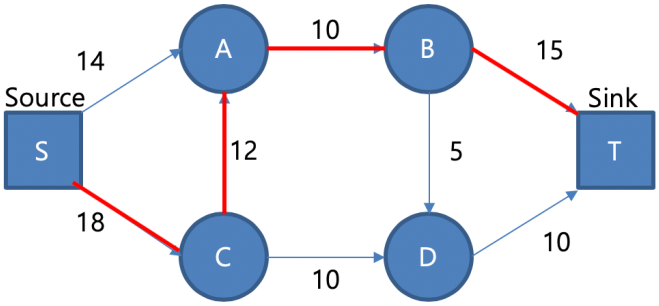
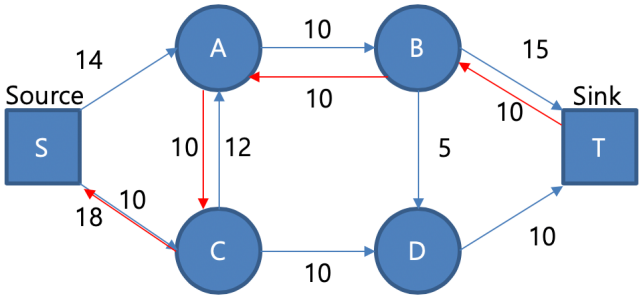
알고리즘

과제번호	14주차
날 짜	2018.12.18
학 번	201302395
이 름	류경빈

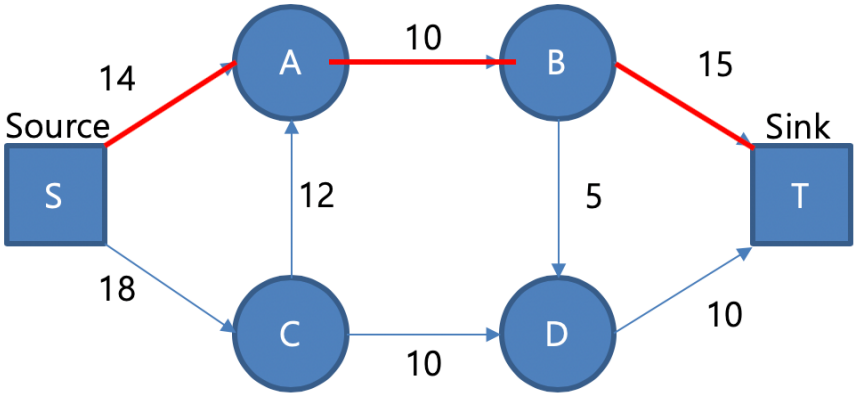
과제 1	Ford Fulkerson

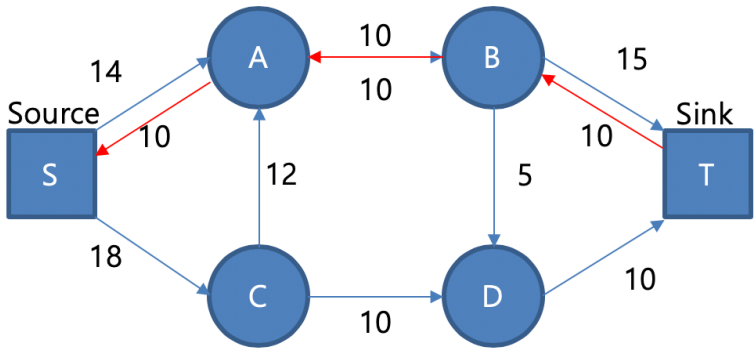




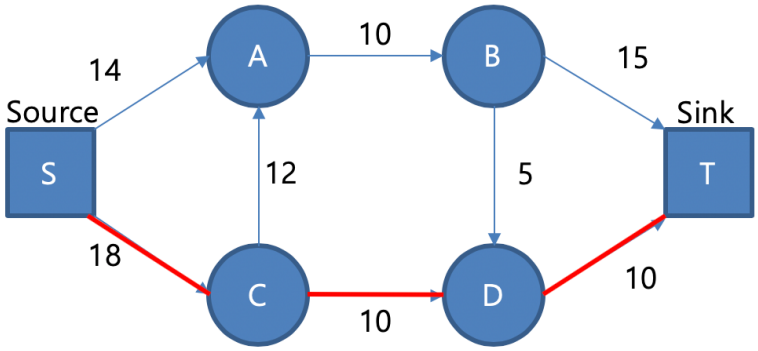


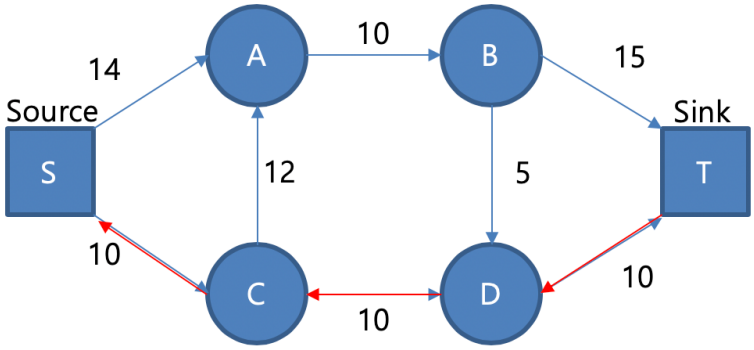
Path A의 최대 유량 : 10



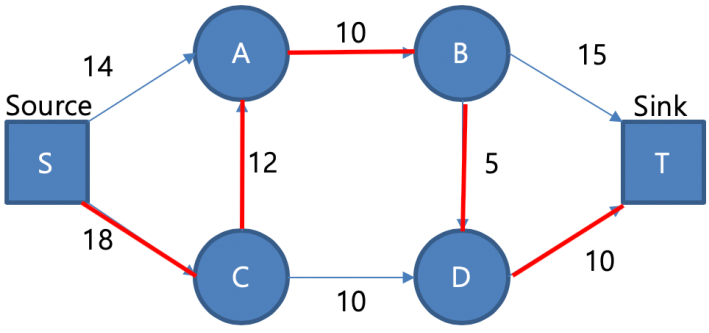


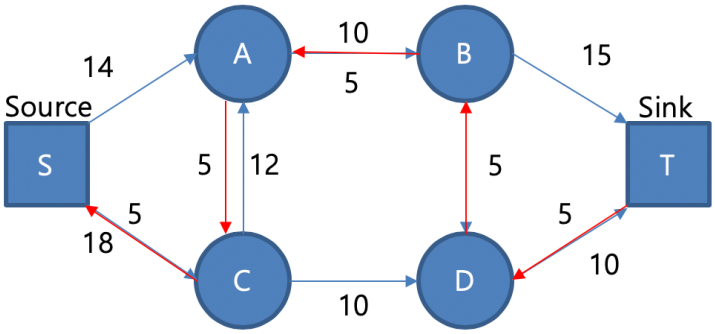
Path B의 최대 유량 : 10





Path C의 최대 유량 : 10



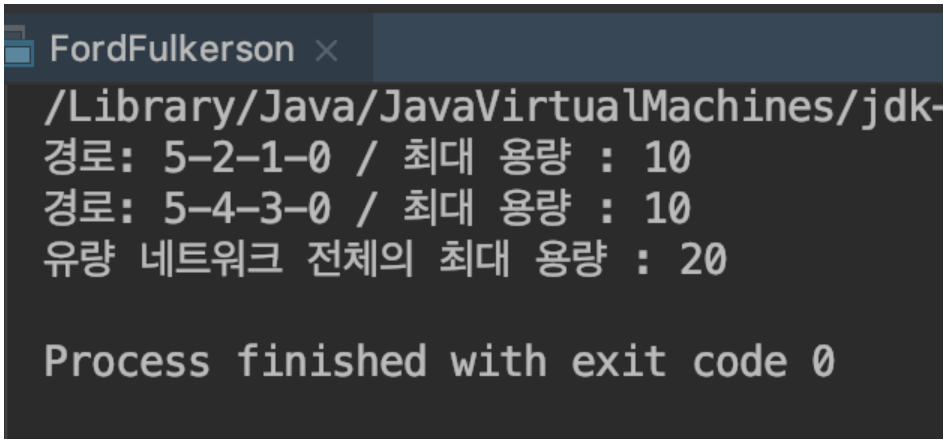


Path D의 최대 유량 : 5

```java
public static int networkFlow(int source, int sink) {
    for (int i = 0; i < V; i++){
        for (int j = 0; j < V; j++){
            flow[i][j] = 0;
        }
    }
    int totalFlow = 0;
    while (true) {
        List<Integer> parent = new ArrayList<>(V);
        for (int i = 0; i < V; i++) {
            parent.add(i,  element: -1);
        }
        Queue<Integer> q = new LinkedList<>();
        parent.set(source, source);
        ((LinkedList<Integer>) q).push(source);
        while (!q.isEmpty() && (parent.get(sink) == -1)) {
            int here = q.peek();
            ((LinkedList<Integer>) q).pop();
            for (int there = 0; there < V; ++there) {
                if (((capacity[here][there] - flow[here][there]) > 0) && parent.get(there) == -1) {
                    q.add(there);
                    parent.set(there, here);
                }
            }
        }
        if (parent.get(sink) == -1) break;

        int amount = INF;

        System.out.print("경로: ");
        for (int p = sink; p != source; p = parent.get(p)) {
            amount = Math.min(capacity[parent.get(p)][p] - flow[parent.get(p)][p], amount);
            System.out.print(p + "-");
        }
        System.out.print("0 / ");

        for (int p = sink; p != source; p = parent.get(p)) {
            flow[parent.get(p)][p] += amount;
            flow[p][parent.get(p)] -= amount;
        }
        totalFlow += amount;
        System.out.println("최대 용량 : " + amount);
    }
    return totalFlow;
}
```

networkFlow 구현

결과 화면


FordFulkerson
/Library/Java/JavaVirtualMachines/jdk-
경로: 5-2-1-0 / 최대 용량 : 10
경로: 5-4-3-0 / 최대 용량 : 10
유량 네트워크 전체의 최대 용량 : 20
Process finished with exit code 0