

알고리즘설계 HW #3

BFS, DFS / Bellman ford

보고서 작성 서약서

1. 나는 타학생의 보고서를 베끼거나 여러 보고서의 내용을 짜집기하지 않겠습니다.

2. 나는 보고서의 주요 내용을 인터넷사이트 등을 통해 얻지 않겠습니다.

3. 나는 보고서의 내용을 조작하지 않겠습니다.

4. 나는 보고서 작성에 참고한 문헌의 출처를 밝히겠습니다.

5. 나는 나의 보고서를 제출 전에 타학생에게 보여주지 않겠습니다.

나는 보고서 작성시 윤리에 어긋난 행동을 하지 않고 정보통신공학인으로서 나의 명예를 지킬 것을 맹세합니다.

2020년 6월 14일

학부 정보통신공학과

학년 4

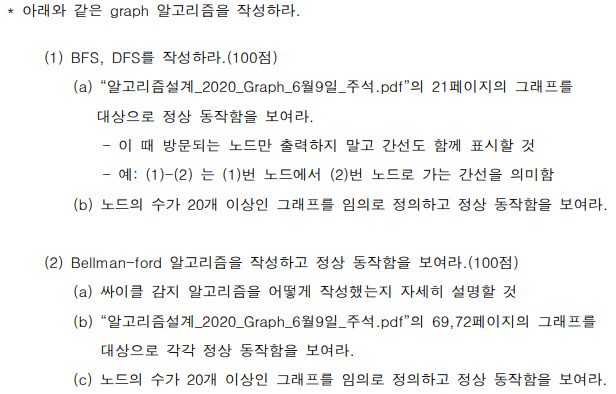
성명 이정우

학번 12171833

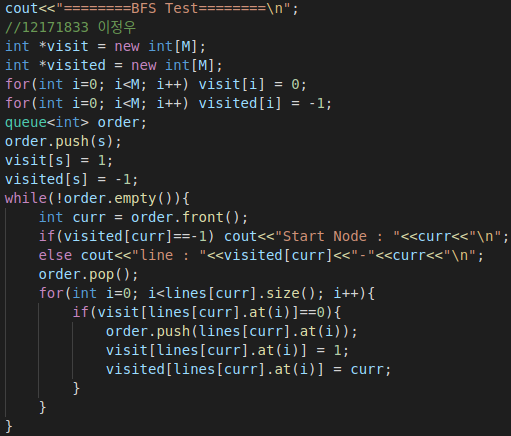


최소한 아래의 내용을 포함할 것.

1. 개요



1. 상세 설계내용
   * + - 1. BFS, DFS를 작성하라.

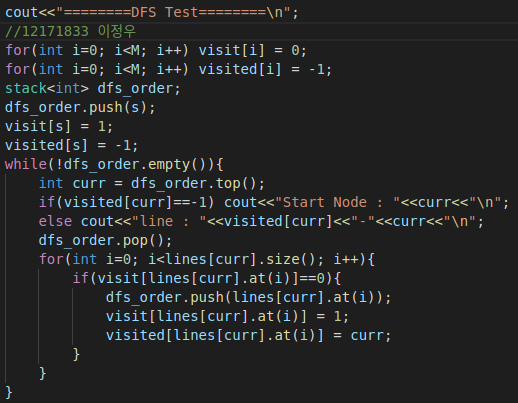


맨 처음 visit, visited배열을 초기화 해준다.

시작위치 s를 큐에 넣어준다.

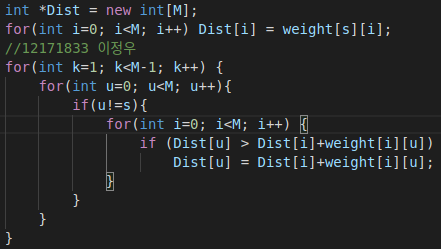
큐에서 꺼낸 위치에 방문하고, 방문한 위치와 이전위치를 출력해준다.

방문한 위치의 값을 큐에서 꺼내고, 현재 방문한 노드에 연결되어 있는 노드를 큐에 넣는다.



큐 대신 스택에 값을 넣어 깊이 우선으로 탐색하도록 한다.

* + - * 1. Bellman-ford 알고리즘을 작성하고 정상 동작함을 보여라.

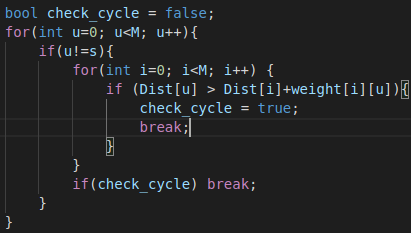


맨 처음에 시작 노드에서 연결된 노드의 가중치를 Dist에 저장해준다.

M-2회 동안 더 작은 가중치로 Dist를 업데이트 해준다.

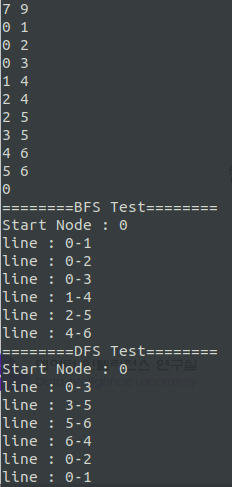
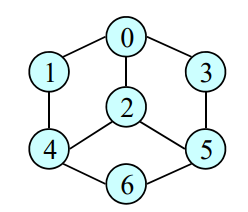
(a) 싸이클 감지 알고리즘을 어떻게 작성했는지 자세히 설명할 것

1)

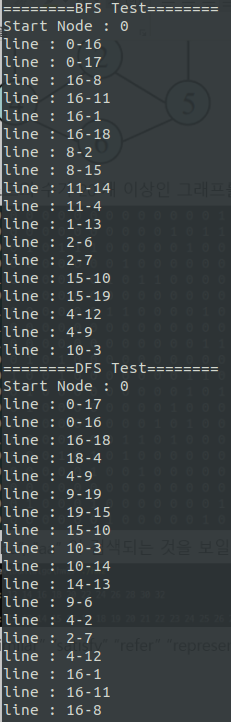
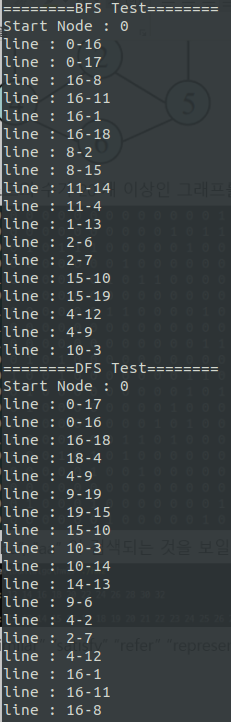
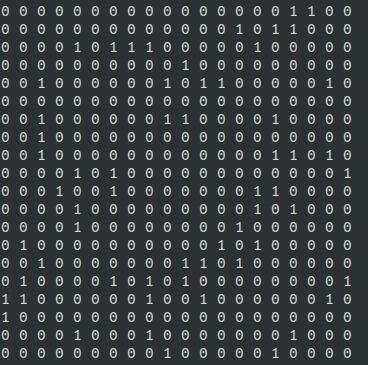


만약 negative cycle을 가지고 있다면 알고리즘 수행 후에 Dist값에 변화가 있을 것이기 때문에 Dist값의 변화를 체크하여 negative cycle을 감지하였다.

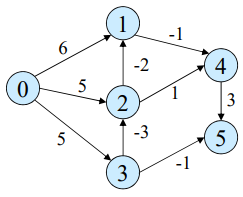
1. 실행 화면

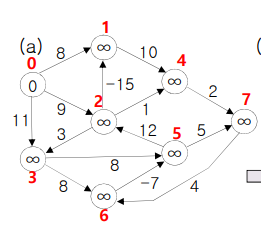
(a) “알고리즘설계\_2020\_Graph\_6월9일\_주석.pdf”의 21페이지의 그래프를 대상으로 정상 동작함을 보여라.

(b) 노드의 수가 20개 이상인 그래프를 임의로 정의하고 정상 동작함을 보여라.

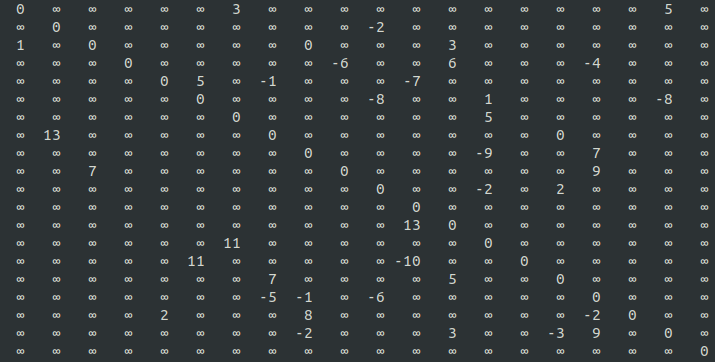


(b) “알고리즘설계\_2020\_Graph\_6월9일\_주석.pdf”의 69,72페이지의 그래프를 대상으로 각각 정상 동작함을 보여라.



(c) 노드의 수가 20개 이상인 그래프를 임의로 정의하고 정상 동작함을 보여라





1. 분석 및 결론

BFS와 DFS알고리즘은 방법이 여러가지가 있었는데, vector를 이용하여 문제를 해결해 보았다. Vector가 아닌 배열을 이용해 문제를 해결하는 것도 가능하였지만 연산할 때, vector를 이용하면 연결되어 있는 간선의 방문 여부만 판단하지만 배열을 이용할 경우 모든 배열 값에 대해 배열의 값이 1인지와 방문 여부를 체크해야 하기 때문에 좀더 오래 걸릴 것이라고 생각하여 vector를 이용하였다.

Bellman-ford알고리즘을 작성하는 것은 어렵지 않았지만 negative cycle을 체크하는 것이 어려웠다. 재귀를 이용하는 등 여러 방법을 생각해보았는데, 결국 negative cycle이 있다는 것은 bellman-ford알고리즘을 모두 수행한 이후에도 negative값으로 Dist배열의 값이 업데이트 된다는 것을 이용하였다. 때문에 알고리즘 수행이후 Dist배열을 한번 더 업데이트 하여 값이 바뀐다면 negative cycle을 가지고 있는 것이라고 판단하였다.