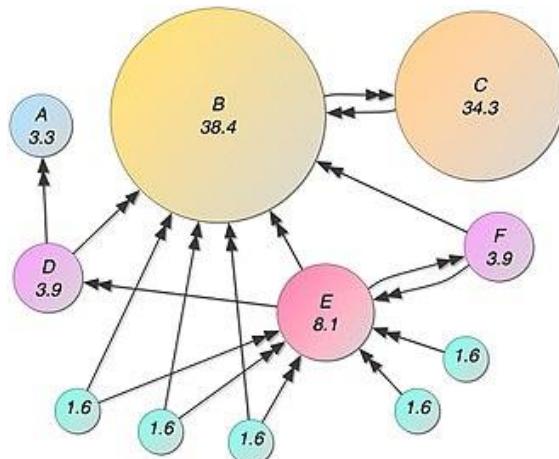


# 자료구조

Midterm Project

PageRank



학 과 : AI 융합학부

이 름 : 20170404 한종수

제출일 : 2021. 06. 05

## 1. 자료 구조

### A. 가중치가 없는 경우

#### i. Struct node

```
static std::mt19937 gen = std::mt19937((unsigned int)time(NULL));
static std::uniform_real_distribution<> hundred(0, 99);
using namespace std;

namespace NoWeight
{
    struct node
    {
        int n_visited;           // No. of visited this node
        int n_neighbors;         // No. of neighbors
        int id;                  // node identifier
        float pageRankVal;      // Current node PageRankVal
        string name;             // string name of this node
        vector<node *> neighbors; // neighbors node address container
    };
}
```

- 각 노드 별로 PageRank를 계산하고, 그 결과 값을 저장하기 위해 이 노드를 몇 번 방문했는지에 대한 n\_visited 변수, 실제 이 노드의 PageRank 값을 저장하는 pageRankVal 변수를 설정하였다.
- 노드를 파일에서 불러올 때, 파일 내에서 노드의 id 가 있고, 이름이 있다. 각각을 id 와 name 변수에 저장한다.
- 그래프를 인접리스트로 구성하는 것이 좋을 것 같아 각 노드마다 이웃들을 저장하고 있는 vector<node \*> neighbors 변수를 설정하였다. 노드 포인터를 저장함으로서 이웃들의 노드로 바로 이동할 수 있다. 이웃들의 수를 n\_neighbors 변수에 저장한다. 이는 RandomWalker 가 이동할 때 확률 계산에 사용된다.

#### ii. Class Graph

```
class Graph
{
    int nodeNum;           // No. of nodes
    vector<node *> nodes; // vector container for nodes
    vector<int> id2IdxTable; // to find index from file node id. first: id from file, second : idx from vector

public:
    Graph(string filename); // Constructor
    ~Graph();
    void LoadEdge(string filename, bool inter); // load edge file
    void AddEdge(int source, int target, bool inter); // adding edge
    int Id2Idx(int n); // convert file node id to vector node id
    void RandomWalker(int i, int n, float q); // i : start node, n : walk length, q : probability jumping to random node
    int PageRank(int i, int n, float q); // cal pagerank
};
```

- 현재 관리되고 있는 그래프의 총 노드 수를 nodeNum 에 저장한다.
- 그래프 노드 전체를 노드 포인터로 관리하는 연결리스트 vector<node \*> nodes 를 정의한다.
- node 파일은 id 순서대로 저장되어있지 않기 때문에, 파일에서 읽은 노드의 id 와 벡터에 저장되는 인덱스가 일치하지 않는다. 파일에서 읽은 노드의 id 로 벡터상에서 인덱스를 찾기 위한 테이블이다.
- Graph(string filename); 노드가 들어있는 파일 이름을 받아서 파일을 읽는다. 파일을 읽어 파일의 노드에 대해서 id 와 이름을 파싱해 node 변수를 초기화 하고, 클래스의 변수 vector<node \*> nodes 에 추가한다. 그리고 전체 노드 수를 증가시킨다. id2IdxTable 에 id 를 읽는 순서대로 저장한다.
- void LoadEdge(string filename, bool inter); edge 가 들어있는 파일 이름을 받아서 파일을 읽는다. 파일 한 줄을 읽어 시점, 종점, 가중치 값으로 나눈다. 각 값으로 AddEdge 함수를 호출한다. 파일이 끝날 때까지 읽는다.
- void AddEdge(int source, int target, bool inter); 시점, 종점, 가중치를 가진 edge 를 얻으면 시점에 해당하는 node 를 nodes 에서 찾아 그 neighbors 변수에 종점 노드를 넣는다. 그리고 n\_neighbors 를 증가시킨다. 만약 방향성이 있는 경우 inter 값을 false 를 넣어주면 앞의 과정까지만 한다. 만약 방향성이 없는 경우, 종점 노드의 이웃에 시점 노드가 있는지 확인 후 없는 경우에 종점 노드의 이웃에 시점 노드를 넣는다. 그리고 이웃의 n\_neighbors 를 증가시킨다.

7. int Id2Idx(int n); 데이터에서 노드들이 id 순으로 저장 되어있지 않음을 확인하였다. 하지만 파일은 순차적으로 읽을 수밖에 없다. 그럼 파일을 읽는 순서대로 노드를 추가한다. 이렇게 되면 파일에서 읽은 노드의 id로 nodes 변수 상에서 특정 노드로 매칭할 수 없게 된다. 이 함수에서는 생성자에서 초기화 해둔 table에서 인자로 들어온 n을 찾아 그 인덱스를 리턴한다.

8. void RandomWalker(int i, int n, float q); 와 int PageRank(int i, int n, float q); 는 뒤에서 설명한다.

## B. 가중치가 있는 경우

### i. Struct node

```
static std::mt19937 gen = std::mt19937((unsigned int)time(NULL));
static std::uniform_real_distribution<> hundred(0, 99);
using namespace std;

namespace Weight
{
    struct node
    {
        int n_visited; // No. of visited this node
        int n_neighbors; // No. of neighbors
        int id; // node identifier
        int sum_weight; // sum of to neighbor weight
        float pageRankVal; // Current node PageRankVal
        string name; // string name of this node
        vector<pair<node *, int>> neighbors; // neighbors node address container
    };
}
```

1. 가중치가 없는 경우와 다른 부분만 작성한다.
2. 그래프를 인접리스트로 구성하는 것이 좋을 것 같아 각 노드마다 이웃들을 저장하고 있는 vector<pair<node \*, int>> neighbors 변수를 설정하였다. 노드 포인터를 저장함으로서 이웃들의 노드로 바로 이동할 수 있다. 이웃 노드와 그로 가는 가중치를 모두 관리할 수 있어야 하므로 std::pair<node\*, int>를 이용하여 둘 변수를 한번에 관리한다. 각 노드가 가진 이웃으로 가는 모든 가중치의 합을 sum\_weight 변수에 저장한다. 이는 RandomWalker가 이동할 때 확률 계산에 사용된다.

### ii. Class Graph

```
class Graph
{
    int nodeNum; // No. of nodes
    vector<node *> nodes; // vector container for nodes
    vector<int> id2IdxTable; // to find index from file node id. first: id from file, second : idx from vector

public:
    Graph(string filename); // Constructor
    ~Graph();
    void LoadEdge(string filename, bool inter); // load edge file
    void AddEdge(int source, int target, int weight, bool inter); // adding edge
    int Id2Idx(int n); // convert file node id to vector node id
    void RandomWalker(int i, int n, float q); // i : start node, n : walk length, q : probability jumping to random node
    int PageRank(int i, int n, float q); // cal pagerank
};
```

1. void AddEdge(int source, int target, int weight, bool inter); 시점, 종점, 가중치를 인자로 받아 시점의 neighbors에 종점을 그로 가는 가중치와 함께 쌍으로 저장한다. 그리고 가중치를 node 내의 전체 가중치 변수에 더한다. inter 변수가 true인 경우 종점에 대해서 종점의 이웃에 시점이 없는 경우 마찬가지로 저장한다.

2. void RandomWalker(int i, int n, float q); 와 int PageRank(int i, int n, float q); 는 뒤에서 설명한다.

## 2. 해야할 일들

### A. (10-11 주차) 가중치 없는 그래프에서 PageRank

#### i. Task1: 가중치가 없는 그래프에서 Random Walker 를 만들어 주세요.

```
//////////////// Task 1 //////////////////
//////////////// implement RandomWalker //////////////////
/////////////////////////////// //////////////////

//todo
// i : start node, n : walk length, q : probability jumping to random node
void Graph::RandomWalker(int i, int n, float q)
{
    int random = 0;           // random value for random jump q
    int random_node = 0;      // random value to pick neighbor to jump
    int q_percent = q * 100; //invert q into percent value
    node *ptr;               // to point current or next node
    ptr = nodes[i];          // ptr initially point start node

    while (n > 0) // loop until end
    {
        if (ptr->n_neighbors == 0) // if no neighbor node, search another node
        {
            std::uniform_real_distribution<> dist(0, nodeNum - 1);
            random_node = dist(gen);
            ptr = nodes[random_node];
            continue;
        }
        random = hundred(gen);
        if ((random <= q_percent) && (q_percent != 0)) //random node pick
        {
            std::uniform_real_distribution<> dist(0, nodeNum - 1);
            random_node = dist(gen);
            ptr = nodes[random_node];
        }
        else // pick node from neighbors
        {
            std::uniform_real_distribution<> dist(0, ptr->n_neighbors - 1);
            random_node = dist(gen);
            ptr = ptr->neighbors[random_node];
        }
        ptr->n_visited++;
        n--;
    }
}
```

- 처음 시작할 노드 번호 i, 실행할 횟수 n, 이웃이 아닌 그래프 상의 random node 로 점프할 확률 q 를 인자로 받는다. Node pointer 변수 ptr 에 처음 시작할 노드의 주소를 저장한다. 이후 ptr 은 RandomWalker 가 현재 머무르고 있는 node 를 가리키게 된다.
- 만약 ptr 이 가리키는 노드가 아무 이웃 노드를 가리키고 있지 않다면, 해당 노드의 RandomWalker 의 이동 횟수를 차감시키지 않고, pagerank 값을 증가시키지 않기 위해 노드의 방문 횟수를 증가시키지 않고 그래프 내의 아무 노드를 선택하여 해당 노드로 지금 노드를 탈출한다.
- Random 적으로 점프하는 확률 q 가 주어진다. 이는 퍼센트가 아닌 1 이하의 소수점 수로 들어온다고 가정한다. 이를 퍼센트로 저장해 q\_percent 변수에 저장한다.
- 이동할 때 이웃 노드로 이동할지 아니면, 그래프 내의 아무 노드로 이동할지는 확률 q\_percent 에 의존한다. q\_percent 는 0~100 의 값이다. 그에 대응하도록 0 ~ 99 의 수를 random 하게 얻어 random 변수에 저장한다. random 변수가 0 ~ 99 의 어떤 정수를 얻을 확률은 각 정수마다 1/100 으로 동일하다.
- 확률적으로 jump 하는 법을 예를 들어 설명하면, 0 ~ 69 의 아무 숫자를 얻을 확률은 0 ~ 69 를 얻을 확률을 모두 더한 값인 70/100 임을 알 수 있다. 그렇다면 random 변수를 q\_percent 이하로 이하로 얻을 확률도 q\_percent / 100 이 된다. 이를 이용하여 조건문에 random <= q\_percent 를 넣으면 q\_percent 확률로 그래프 내의 랜덤한 노드로 이동하는 동작을 수행하게 할 수 있다. 단, q\_percent 가 0 인 경우는 확률이 0 이므로 점프하지 않는다.
- 그래프 내의 랜덤한 노드로 이동하는 동작을 수행하는 경우에 0 ~ nodeNum - 1 의 수를 random 하게 얻어 random\_node 에 저장한다. 이는 Class Graph 의 변수 nodes 에서 node 를 얻는 index 로 사용된다. Nodes[random\_node]를 통해 다음 이동할 노드의 주소를 얻고 이를 ptr 에 연결시킨다. 방문할 ptr 의 방문횟수를 증가시키고 노드의 이동 횟수를 차감시킨다. 다음 루프를 진행한다.
- 현재 노드의 이웃 노드로 이동하는 경우 0 ~ n\_neighbors - 1 의 수를 얻어 random\_node 에 저장한다. 이는 현재 노드가 가리키는 노드를 저장하는 변수 neighbors 에서 node 를 얻는 index 로 사용된다. 다음 진행할 노드의 주소를 ptr->neighbors[random\_node]로 구할 수 있다. 이를 ptr 로 연결시킨다. 방문할 ptr 의 방문횟수를 증가시키고 노드의 이동 횟수를 차감시킨다. 다음 루프를 진행한다.
- 각 변수의 수치에 대한 내용은 아래 실험에 따로 첨부하였습니다!

## ii. Task2: 이를 통해서 PageRank 를 구하는 함수를 만들어 주세요.

```
////////// Task 2 //////////
//////// implement PageRank func //////////
//////////
```

```
// todo
// i : start node, n : walk length, q : probability jumping to random node
int Graph::PageRank(int i, int n, float q)
{
    // First, call RandomWalker
    vector<pair<float, int>> rank;
    RandomWalker(i, n, q);

    // Second, iterate all nodes and calculate pagerank
    for (vector<node *>::iterator it = nodes.begin(); it != nodes.end(); it++)
    {
        (*it)->pageRankVal = (float)((*it)->n_visited) / (float)n;
        rank.push_back({(*it)->pageRankVal, (*it)->id});
    }

    // Third, print sorted pagerank
    sort(rank.begin(), rank.end());
    for (vector<pair<float, int>>::iterator it = rank.begin(); it != rank.end(); it++)
    {
        cout << "id : " << (*it).second << "\tpageRankValue : " << (*it).first << "\tname : " << nodes[Id2Idx((*it).second)]->name << endl;
    }
}
```

- 처음 시작할 노드 번호 i, 실행할 횟수 n, 이웃이 아닌 그래프 상의 random node 로 점프할 확률 q 를 인자로 받아 내부에서 RandomWalker 를 실행한다.
- RandomWalker 가 각 노드를 방문한 횟수가 각 노드의 n\_visited 변수에 저장되었다. 각 노드는 Class Graph 의 변수인 nodes 를 탐색하며 방문할 수 있다.
- 어떤 노드에 대한 PageRank 는 그 노드를 RandomWalker 가 방문한 횟수에 RandomWalker 의 전체 이동 횟수를 나눔으로써 구할 수 있다. 이를 식으로 표현하면, nodes->n\_visited / n 이 된다.
- 위의 방법을 전체 노드에 적용하면, 전체 노드를 탐색하는 방법으로 iterator 을 사용함으로 각 노드의 pagerank 는 (\*it)->n\_visited / n; 으로 계산되고 결과를 각 노드의 pageRankVal 변수에 저장한다. 모든 노드에 대해서 pagerank 를 계산해 저장한다.
- Pagerank 를 보기 좋게 하기 위해 pagerank 값으로 정렬시킨다. 빠르게 정렬시키기 위해 pagerank 와 id 값만을 저장하는 pair 쌍을 저장하는 벡터를 정의한 후 pagerank 값에 대해 오름차순 정렬시킨다.
- pair 쌍을 저장하는 벡터 rank 를 순회하며 pagerank 가 낮은 노드부터 id 와 pagerank 값, 이름을 차례로 출력한다.

## iii. Task3: 실제 그래프 데이터에 대해서 PageRank 를 수행해보세요.

id : 837	pageRankValue : 0.000647967	name : 한남나들목 입구
id : 227	pageRankValue : 0.000651167	name : 양평2나들목 보행동로 입구
id : 1911	pageRankValue : 0.0006536	name : 구로디지털단지역 앞
id : 765	pageRankValue : 0.000654133	name : 오목교역 3번 출구
id : 274	pageRankValue : 0.000657733	name : 영등포역지하쇼핑센타 5번 출구
id : 2645	pageRankValue : 0.000660533	name : 잠실나들목2(엘스아파트136동)
id : 2025	pageRankValue : 0.000664667	name : 흑석역 1번 출구
id : 361	pageRankValue : 0.000668933	name : 동묘앞역 1번 출구 뒤
id : 829	pageRankValue : 0.000669567	name : 베르가모앞
id : 2525	pageRankValue : 0.000671067	name : 반포소평타운 2동 앞
id : 146	pageRankValue : 0.0006731	name : 마포역 2번 출구 뒤
id : 1209	pageRankValue : 0.000674833	name : 잠실나루역 (2번 출구 쪽)
id : 144	pageRankValue : 0.000676267	name : 공덕역 8번 출구
id : 114	pageRankValue : 0.0006794	name : 홍대입구역 8번 출구 앞
id : 2102	pageRankValue : 0.000682933	name : 봉림교 교통섬
id : 853	pageRankValue : 0.000684133	name : 용산역 맞은편
id : 2301	pageRankValue : 0.000713267	name : 현대고등학교 건너편
id : 1210	pageRankValue : 0.000739833	name : 롯데월드타워(잠실역 2번 출구 쪽)
id : 1295	pageRankValue : 0.000739867	name : 잠실역 8번 출구
id : 2262	pageRankValue : 0.000744167	name : 한신16차아파트 119동 앞
id : 210	pageRankValue : 0.0007475	name : IFC몰
id : 3511	pageRankValue : 0.0007605	name : 응봉역 1번 출구
id : 272	pageRankValue : 0.000781067	name : 당산육갑문
id : 1906	pageRankValue : 0.000793633	name : 신도림역 1번 출구 앞
id : 186	pageRankValue : 0.000794067	name : 합정역 7번 출구 앞
id : 152	pageRankValue : 0.000801067	name : 마포구민체육센터 앞
id : 825	pageRankValue : 0.0008013	name : 서빙고동 주민센터 앞
id : 583	pageRankValue : 0.000903367	name : 청계천 생태교실 앞
id : 2219	pageRankValue : 0.000903733	name : 고속터미널역 8-1번, 8-2번 출구 사이
id : 502	pageRankValue : 0.000984233	name : 뚝섬유원지역 1번 출구 앞
id : 207	pageRankValue : 0.000985	name : 여의나루역 1번 출구 앞
id : 565	pageRankValue : 0.000998	name : 옥수역 3번 출구

1. bicycle\_trips\_all.tsv

PageRank(50, 30000000, 0.7);

id : 60	pageRankValue : 0.00892823	name : MON MOTHMA
id : 83	pageRankValue : 0.00897123	name : PIETT
id : 53	pageRankValue : 0.0089806	name : COUNT DOOKU
id : 77	pageRankValue : 0.0090726	name : RED LEADER
id : 69	pageRankValue : 0.00926993	name : BIGGS
id : 48	pageRankValue : 0.0094535	name : BOBA FETT
id : 32	pageRankValue : 0.00960353	name : MACE WINDU
id : 99	pageRankValue : 0.00968117	name : REY
id : 33	pageRankValue : 0.00996217	name : KI-ADI-MUNDI
id : 97	pageRankValue : 0.0100991	name : FINN
id : 2	pageRankValue : 0.0103036	name : BB-8
id : 29	pageRankValue : 0.0103666	name : JABBA
id : 76	pageRankValue : 0.0108946	name : WEDGE
id : 95	pageRankValue : 0.0110444	name : KYLO REN
id : 100	pageRankValue : 0.0111541	name : GENERAL HUX
id : 36	pageRankValue : 0.0112198	name : BAIL ORGANA
id : 1	pageRankValue : 0.011304	name : CHEWBACCA
id : 94	pageRankValue : 0.0118374	name : POE
id : 0	pageRankValue : 0.0119102	name : R2-D2
id : 34	pageRankValue : 0.012005	name : YODA
id : 4	pageRankValue : 0.0128325	name : NUTE GUNRAY
id : 11	pageRankValue : 0.0131334	name : EMPEROR
id : 73	pageRankValue : 0.0144175	name : HAN
id : 66	pageRankValue : 0.0144197	name : DARTH VADER
id : 3	pageRankValue : 0.0144881	name : QUI-GON
id : 70	pageRankValue : 0.0145564	name : LEIA
id : 14	pageRankValue : 0.0156573	name : JAR JAR
id : 24	pageRankValue : 0.0172094	name : C-3PO
id : 17	pageRankValue : 0.0175636	name : PADME
id : 67	pageRankValue : 0.0177715	name : LUKE
id : 7	pageRankValue : 0.020954	name : OBI-WAN
id : 20	pageRankValue : 0.0216744	name : ANAKIN

2. starwars—full—interactions—allCharacters-links.tsv      PageRank(50, 30000000, 0.7);

## B. (12 주차) 가중치 있는 그래프에서

### i. Task4: 가중치가 있는 그래프에서 Random Walker 를 만들어 주세요.

```
///////////////////////////////
// Task 4 //////////////////
//////////////// implement RandomWalker //////////////////
///////////////////////////////



//todo
// i : start node, n : walk length, q : probability jumping to random node
void Graph::RandomWalker(int i, int n, float q)
{
    int random = 0;           // random value for random jump q
    int random_node = 0;      // random value to pick neighbor to jump
    int q_percent = q * 100; // invert q into percent value
    node *ptr;                // to point current or next node
    ptr = nodes[i];           // ptr initially point start node

    int sum = 0;
    vector<pair<node *, int>>::iterator it;

    while (n > 0) // loop until end
    {
        if (ptr->n_neighbors == 0) // if no neighbor node, search another node
        {
            std::uniform_real_distribution<> dist(0, nodeNum - 1);
            random_node = dist(gen);
            ptr = nodes[random_node];
            continue;
        }
        random = hundred(gen);
        if ((random <= q_percent) && (q_percent != 0)) //random node pick
        {
            std::uniform_real_distribution<> dist(0, nodeNum - 1);
            random_node = dist(gen);
            ptr = nodes[random_node];
        }
        else // pick node from neighbors
        {
            // get 0 <= random_value < sum_weight
            // keep sum neighbor nodes weight until random_node
            // if sum > random_node, the node being about to sum is randomly chosen node.
            sum = ptr->neighbors[0].second;
            std::uniform_real_distribution<> dist(0, ptr->sum_weight);
            random_node = dist(gen);

            for (it = ptr->neighbors.begin() + 1; (it != ptr->neighbors.end()) && (sum < random_node); it++)
            {
                sum += (*it).second;
            }
            if (it == ptr->neighbors.end())
            {
                it--;
                ptr = (*it).first;
            }
            ptr->n_visited++;
            n--;
        }
    }
}
```

1. 처음 시작할 노드 번호 i, 실행할 횟수 n, 이웃이 아닌 그래프 상의 random node로 점프할 확률 q를 인자로 받는다. Node pointer 변수 ptr에 처음 시작할 노드의 주소를 저장한다. 이후 ptr은 RandomWalker가 현재 머무르고 있는 node를 가리키게 된다.
2. 만약 ptr이 가리키는 노드가 아무 이웃 노드를 가리키고 있지 않다면, 해당 노드의 RandomWalker의 이동 횟수를 차감시키지 않고, pagerank 값을 증가시키지 않기 위해 노드의 방문 횟수를 증가시키지 않고 그래프 내의 아무 노드를 선택하여 해당 노드로 지금 노드를 탈출한다.
3. Random 적으로 점프하는 확률 q가 주어진다. 이는 퍼센트가 아닌 1 이하의 소수점 수로 들어온다고 가정한다. 이를 퍼센트로 저장해 q\_percent 변수에 저장한다.
4. 이동할 때 이웃 노드로 이동할지 아니면, 그래프 내의 아무 노드로 이동할지는 확률 q\_percent에 의존한다. q\_percent는 0~100의 값이다. 그에 대응하도록 0 ~ 99의 수를 random하게 얻어 random 변수에 저장한다. random 변수가 0 ~ 99의 어떤 정수를 얻을 확률은 각 정수마다 1/100으로 동일하다.
5. 확률적으로 jump 하는 법을 예를 들어 설명하면, 0 ~ 69의 아무 숫자를 얻을 확률은 0 ~ 69를 얻을 확률을 모두 더한 값인 70/100임을 알 수 있다. 그렇다면 random 변수를 q\_percent 이하로 이하로 얻을 확률도 q\_percent / 100이 된다. 이를 이용하여 조건문에 random <= q\_percent를 넣으면 q\_percent 확률로 그래프 내의 랜덤한 노드로 이동하는 동작을 수행하게 할 수 있다. 단, q\_percent가 0인 경우는 확률이 0이므로 점프하지 않는다.
6. 그래프 내의 랜덤한 노드로 이동하는 동작을 수행하는 경우에 0 ~ nodeNum - 1의 수를 random하게 얻어 random\_node에 저장한다. 이는 Class Graph의 변수 nodes에서 node를 얻는 index로 사용된다. Nodes[random\_node]를 통해 다음 이동할 노드의 주소를 얻고 이를 ptr에 연결시킨다. 방문할 ptr의 방문횟수를 증가시키고 노드의 이동 횟수를 차감시킨다. 다음 루프를 진행한다.
7. 가중치를 가진 상황에서 이웃한 노드로 이동하는 경우는 가중치가 없는 경우와 그 방법이 다르다. 원리는 q의 확률로 jump했던 원리와 비슷하다. 각 노드는 이웃 노드로 가는 가중치를 모두 더한 값 sum\_weight가 있다. 현재 노드의 이웃 노드로 이동하는 경우 1 ~ sum\_weight의 수를 얻어 random\_node에 저장한다. 범위 내의 각 정수를 얻을 확률은 모두 1 / sum\_weight로 같다.
8. 첫 번째 이웃 노드의 가중치를 sum에 저장한다. 그리고 반복문을 도는데, 반복문에서 시작을 두 번째 이웃 노드로 한다. 그리고 조건을 마지막 이웃을 방문한 후와 sum이 random\_node를 넘기 전까지로 한다. sum이 random\_node 이상인 경우 loop를 탈출한다.
9. 예를 들어 만약 가장 처음 노드의 weight가 3이면, random\_node 값이 1~3인 경우에 루프를 돌지 않고 탈출한다. 그럼 가장 처음 노드가 선택될 확률은 3 \* 1 / sum\_weight로 볼 수 있다. 두 번째 노드의 weight가 2이고 random\_node가 4이상인 경우, loop를 한번 돌게 되고 sum은 5가 된다. random\_node 값이 4~5인 경우에 두 번째 노드가 선택되고 그 확률은 2 \* 1 / sum\_weight이다. 만약 sum\_weight과 random\_node가 같은 경우 마지막 노드까지 방문할 수 있다. 이런 방식으로 다음 이동할 노드를 가중치를 통한 확률을 통해 정할 수 있다. 그렇게 정해진 다음 노드를 ptr로 연결시킨다. 방문할 ptr의 방문횟수를 증가시키고 노드의 이동 횟수를 차감시킨다. 다음 루프를 진행한다.
10. 각 변수의 수치에 대한 내용은 아래 실험에 따로 첨부하였습니다!

## ii. Task5: 이를 통해 가중치 있는 그래프에서 PageRank를 구하는 함수를 만들어 주세요

```
////////////////// Task 5 //////////////////
////////////////// implement PageRank func //////////////////

//todo
// i : start node, n : walk length, q : probability jumping to random node
int Graph::PageRank(int i, int n, float q)
{
    // First, call RandomWalker
    vector<pair<float, int>> rank;
    RandomWalker(i, n, q);

    // Second, iterate all nodes and calculate pagerank
    for (vector<node *>::iterator it = nodes.begin(); it != nodes.end(); it++)
    {
        (*it)->pageRankVal = (float)((*it)->n_visited) / (float)n;
        rank.push_back((*it)->pageRankVal, (*it)->id);
    }

    // Third, print sorted pagerank
    sort(rank.begin(), rank.end());
    for (vector<pair<float, int>>::iterator it = rank.begin(); it != rank.end(); it++)
    {
        cout << "id : " << (*it).second << "\tpageRankValue : " << (*it).first << "\tname : " << nodes[Id2Idx((*it).second)]->name << endl;
    }
}
```

- 처음 시작할 노드 번호  $i$ , 실행할 횟수  $n$ , 이웃이 아닌 그래프 상의 random node로 점프할 확률  $q$ 를 인자로 받아 내부에서 RandomWalker를 실행한다.
- RandomWalker가 각 노드를 방문한 횟수가 각 노드의  $n_{visited}$  변수에 저장되었다. 각 노드는 Class Graph의 변수인 nodes를 탐색하며 방문할 수 있다.
- 어떤 노드에 대한 PageRank는 그 노드를 RandomWalker가 방문한 횟수에 RandomWalker의 전체 이동 횟수를 나눔으로써 구할 수 있다. 이를 식으로 표현하면,  $nodes \rightarrow n_{visited} / n$  이 된다.
- 위의 방법을 전체 노드에 적용하면, 전체 노드를 탐색하는 방법으로 iterator를 사용함으로 각 노드의 pagerank는  $(\ast it) \rightarrow n_{visited} / n$ ;으로 계산되고 결과를 각 노드의 pageRankVal 변수에 저장한다. 모든 노드에 대해서 pagerank를 계산해 저장한다.
- Pagerank를 보기 좋게 하기 위해 pagerank 값으로 정렬시킨다. 빠르게 정렬시키기 위해 pagerank와 id값만을 저장하는 pair 쌍을 저장하는 벡터를 정의한 후 pagerank 값에 대해 오름차순 정렬시킨다.
- pair 쌍을 저장하는 벡터 rank를 순회하며 pagerank 가 낮은 노드부터 id와 pagerank 값, 이름을 차례로 출력한다.

### iii. Task6: 실제 가중치가 있는 그래프 데이터에 대해서 PageRank를 수행해보세요

id : 1653	pageRankValue : 0.000734633	name : 노원역 1번출구
id : 2220	pageRankValue : 0.0007359	name : 반포본동 주민센터 앞
id : 2264	pageRankValue : 0.000739133	name : 이수역 1번출구
id : 3533	pageRankValue : 0.000739133	name : 건대입구역 사거리(롯데백화점)
id : 2222	pageRankValue : 0.000739933	name : 사당역 12번출구 뒤
id : 1531	pageRankValue : 0.000742067	name : 미아사거리 1번 출구
id : 2340	pageRankValue : 0.000742567	name : 삼호물산버스정류장(23370) 옆
id : 210	pageRankValue : 0.0007487	name : IFC몰
id : 1009	pageRankValue : 0.000749133	name : 천호역 4번출구(현대백화점)
id : 3511	pageRankValue : 0.000768	name : 응봉역 1번출구
id : 2715	pageRankValue : 0.0007707	name : 마곡나루역 2번 출구
id : 1986	pageRankValue : 0.000778367	name : 신도림역 1번 출구 앞
id : 2183	pageRankValue : 0.000779933	name : 동방1교
id : 1153	pageRankValue : 0.0007822	name : 발산역 1번, 9번 인근 대여소
id : 602	pageRankValue : 0.000782833	name : 장안동 사거리
id : 3010	pageRankValue : 0.0007854	name : 홍대입구역 3번출구
id : 913	pageRankValue : 0.000795967	name : 이마트 은평점
id : 2255	pageRankValue : 0.000806967	name : 시민의숲역 1번출구 옆
id : 1308	pageRankValue : 0.000811933	name : 안암로터리 버스정류장 앞
id : 144	pageRankValue : 0.000812133	name : 공덕역 8번출구
id : 1210	pageRankValue : 0.000838467	name : 롯데월드타워(잠실역2번출구 쪽)
id : 114	pageRankValue : 0.0008433	name : 홍대입구역 8번출구 앞
id : 1911	pageRankValue : 0.0008554	name : 구로디지털단지역 앞
id : 907	pageRankValue : 0.000859133	name : CJ 드림시티
id : 583	pageRankValue : 0.000860067	name : 청계천 생태교실 앞
id : 853	pageRankValue : 0.000867267	name : 용산역 맞은편
id : 2182	pageRankValue : 0.000889367	name : 봉림교 교통섬
id : 502	pageRankValue : 0.000987933	name : 뚝섬유원지역 1번출구 앞
id : 152	pageRankValue : 0.0009911	name : 매포구민체육센터 앞
id : 565	pageRankValue : 0.0010466	name : 옥수역 3번출구
id : 2219	pageRankValue : 0.00108783	name : 고속터미널역 8-1번, 8-2번 출구 사이
id : 207	pageRankValue : 0.0011092	name : 여의나루역 1번출구 앞

- bicycle\_trips\_all.tsv PageRank(50, 30000000, 0.7);

id : 107	pageRankValue : 0.0088909	name : YOLO ZIFF
id : 32	pageRankValue : 0.00893067	name : MACE WINDU
id : 83	pageRankValue : 0.0089741	name : PIETT
id : 23	pageRankValue : 0.00925763	name : SHMI
id : 60	pageRankValue : 0.0092709	name : MON MOTHMA
id : 100		name : GENERAL HUX
id : 94	pageRankValue : 0.00937047	name : POE
id : 11	pageRankValue : 0.00965553	name : EMPEROR
id : 97	pageRankValue : 0.00979907	name : FINN
id : 21	pageRankValue : 0.0100372	name : SEBULBA
id : 10	pageRankValue : 0.0101608	name : TEY HOW
id : 109		name : ELLO ASTY
id : 29	pageRankValue : 0.0102266	name : JABBA
id : 1	pageRankValue : 0.010319	name : CHEWBACCA
id : 76	pageRankValue : 0.0103532	name : WEDGE
id : 33	pageRankValue : 0.0105052	name : KI-ADI-MUNDI
id : 99	pageRankValue : 0.0109663	name : REY
id : 2	pageRankValue : 0.0111501	name : BB-8
id : 34	pageRankValue : 0.0114757	name : YODA
id : 3	pageRankValue : 0.0117609	name : QUI-GON
id : 95	pageRankValue : 0.0123545	name : KYLO REN
id : 69	pageRankValue : 0.0127139	name : BIGGS
id : 4	pageRankValue : 0.0131338	name : NUTE GUNRAY
id : 66	pageRankValue : 0.0137495	name : DARTH VADER
id : 24	pageRankValue : 0.0163321	name : C-3PO
id : 14	pageRankValue : 0.0169979	name : JAR JAR
id : 73	pageRankValue : 0.0174907	name : HAN
id : 70	pageRankValue : 0.0189105	name : LEIA
id : 67	pageRankValue : 0.0199596	name : LUKE
id : 7	pageRankValue : 0.0199786	name : OBI-WAN
id : 17	pageRankValue : 0.0202875	name : PADME
id : 20	pageRankValue : 0.0234333	name : ANAKIN

- starwars-full-interactions-allCharacters-links.tsv PageRank(50, 30000000, 0.7);

#### iv. Task7: 가중치가 있는 경우와 없는 경우가 어떻게 달라지나요? 어떤 것이 더 잘 맞는 것 같나요?

1. 결과는 대체로 비슷한 것 같습니다. 하지만 가중치에 따라서 크게 등수가 변경된 일부 노드들이 있는 것 같습니다. 제 생각에는 가중치가 있는 경우가 더 잘 맞는 것 같습니다.
2. 왜냐하면 정보는 많을수록 좋고, 그 정보를 이용할수록 데이터가 더 정확해진다고 생각하기 때문입니다. 가중치가 없을 때는 이웃 노드 선택 시, 모든 이웃 노드가 같은 가치를 가지지만, 가중치가 있는 경우에는 그 중 가치가 더 높은 노드를 확률적으로 더 높게 선택할 수 있습니다. 이는 가중치가 있는 경우에서 더 중요한 노드에 대해서 다른 노드와 더 뚜렷한 차이를 만들어 주기 때문에 가장 중요한 노드에 대해 그 중요성을 더 명확하게 보여줄 수 있다고 생각합니다.
3. 저는 pagerank에서 중요한 점이 그래프의 노드들 중에서 가장 중요한 노드를 얼마나 명확하게 잘 뽑아줄 수 있느냐 라고 생각합니다. 따라서 저는 가중치가 있는 경우가 더 잘 맞다고 생각합니다.

#### 3. 실험 (data는 bicycle\_trips\_all.tsv, graph는 가중치 graph로 진행하였습니다. 그리고 위의 코드, 결과와 다르게 rank 순위를 보여주도록 pagerank를 수정하였습니다. 실험 변수는 기본 PageRank (50, 30000000, 0.7);로 두고 하였습니다.)

A. 기본 RandomWalker 변수 값에 따른 결과 분석 (저는 맨 처음에 PageRank(0, 200, 0.5)가 충분하다고 생각했었습니다. 실험을 진행해보고 나서 i는 크게 상관 없고, n = 30,000,000, q = 0.15 일 때가 적당하다고 생각합니다.)

##### i. i 값 실험. Rank50 위 까지만 데이터로 사용하였습니다.

rank : 50	pageRankValue : 0.0006945	name : 한신 16차아파트 119동 앞	rank : 50	pageRankValue : 0.000698367	name : 마포구청 역
rank : 49	pageRankValue : 0.000694733	name : 동묘앞역 1번출구 뒤	rank : 49	pageRankValue : 0.000699067	name : 이마트 버스정류소 옆
rank : 48	pageRankValue : 0.000696767	name : 신도림테크노근린공원	rank : 48	pageRankValue : 0.000699933	name : 신도림테크노근린공원
rank : 47	pageRankValue : 0.000699967	name : 장한평역 1번출구 (국민은행앞)	rank : 47	pageRankValue : 0.000701033	name : 서울광장 옆
rank : 46	pageRankValue : 0.000701367	name : 흥사거리	rank : 46	pageRankValue : 0.000701433	name : 신용산역 1번 출구
rank : 45	pageRankValue : 0.000702067	name : 마포구청 역	rank : 45	pageRankValue : 0.000705667	name : 흥은사거리
rank : 44	pageRankValue : 0.000708367	name : 가양역 8번출구	rank : 44	pageRankValue : 0.000708633	name : 망원역 1번출구 앞
rank : 43	pageRankValue : 0.0007123	name : 이디야 커피 수유 역점 앞	rank : 43	pageRankValue : 0.000710967	name : 가양역 8번출구
rank : 42	pageRankValue : 0.000712367	name : 공등역 1번 출구 앞	rank : 42	pageRankValue : 0.000717467	name : 노원역 1번 출구
rank : 41	pageRankValue : 0.000714767	name : 망원역 1번 출구 앞	rank : 41	pageRankValue : 0.000719867	name : 잠실역 8번출구
rank : 40	pageRankValue : 0.000721933	name : 석계역 문화광장 내 자전거 보관소	rank : 40	pageRankValue : 0.000719267	name : 이디야 커피 수유 역점 앞
rank : 39	pageRankValue : 0.000723	name : 당곡사거리	rank : 39	pageRankValue : 0.000719433	name : 석계역 문화광장 내 자전거 보관소
rank : 38	pageRankValue : 0.0007237	name : 마곡나루역 5번출구 뒤편	rank : 38	pageRankValue : 0.000722833	name : 마곡나루역 5번출구 뒤편
rank : 37	pageRankValue : 0.000725667	name : 건대입구역 사거리 (롯데백화점)	rank : 37	pageRankValue : 0.00072433	name : 당곡사거리
rank : 36	pageRankValue : 0.000725733	name : 엔트럴 푸르지오 시티 앞	rank : 36	pageRankValue : 0.000725867	name : 응봉역 1번 출구 앞
rank : 35	pageRankValue : 0.000725767	name : 노원역 1번출구	rank : 35	pageRankValue : 0.000726667	name : 신대방역 2번 출구
rank : 34	pageRankValue : 0.000727833	name : 사당역 12번출구 뒤	rank : 34	pageRankValue : 0.000726767	name : 미아사거리 1번 출구
rank : 33	pageRankValue : 0.000728633	name : 잠실역 8번출구	rank : 33	pageRankValue : 0.000727433	name : 건대입구역 사거리 (롯데백화점)
rank : 32	pageRankValue : 0.000728767	name : 둔촌역 3번 출입구	rank : 32	pageRankValue : 0.0007291	name : 센트럴 푸르지오 시티 앞
rank : 31	pageRankValue : 0.0007296	name : 미아사거리 1번 출구	rank : 31	pageRankValue : 0.000730567	name : 둔촌역 3번 출입구
rank : 30	pageRankValue : 0.000730867	name : 신대방역 2번 출구	rank : 30	pageRankValue : 0.000732867	name : 사당역 12번 출구 뒤
rank : 29	pageRankValue : 0.000737633	name : 이수역 1번출구	rank : 29	pageRankValue : 0.0007379	name : 반포본동 주민센터 앞
rank : 28	pageRankValue : 0.0007414	name : 반포본동 주민센터 앞	rank : 28	pageRankValue : 0.000737967	name : 영등포구청역 1번출구
rank : 27	pageRankValue : 0.0007423	name : 영등포구청역 1번출구	rank : 27	pageRankValue : 0.000741667	name : 이수역 1번출구
rank : 26	pageRankValue : 0.000746667	name : IFC몰	rank : 26	pageRankValue : 0.000745633	name : 천호역 4번출구 (현대백화점)
rank : 25	pageRankValue : 0.0007491	name : 천호역 4번출구 (현대백화점)	rank : 25	pageRankValue : 0.000746367	name : 삼호물산버스정류장 (23370) 옆
rank : 24	pageRankValue : 0.0007499	name : 강남역 1번 출구	rank : 24	pageRankValue : 0.000747478	name : IFC몰
rank : 23	pageRankValue : 0.000766433	name : 응봉역 1번 출구	rank : 23	pageRankValue : 0.000767449	name : 응봉역 1번 출구
rank : 22	pageRankValue : 0.000777067	name : 신도림역 1번 출구 앞	rank : 22	pageRankValue : 0.000773433	name : 발산역 1번, 9번 인근 대여소
rank : 21	pageRankValue : 0.000778367	name : 마곡나루역 2번 출구	rank : 21	pageRankValue : 0.0007736	name : 장안동 사거리
rank : 20	pageRankValue : 0.000782033	name : 강안동 사거리	rank : 20	pageRankValue : 0.000774233	name : 마곡나루역 2번 출구
rank : 19	pageRankValue : 0.000782833	name : 발산역 1번, 9번 인근 대여소	rank : 19	pageRankValue : 0.000779967	name : 신도림역 1번 출구 앞
rank : 18	pageRankValue : 0.000792267	name : 동방1교	rank : 18	pageRankValue : 0.000784333	name : 동방1교
rank : 17	pageRankValue : 0.000793133	name : 홍대입구역 3번출구	rank : 17	pageRankValue : 0.000788467	name : 이마트 옆 평점
rank : 16	pageRankValue : 0.000793533	name : 이마트 옆 평점	rank : 16	pageRankValue : 0.000795867	name : 홍대입구역 3번출구
rank : 15	pageRankValue : 0.000806967	name : 시민의숲역 1번출구 옆	rank : 15	pageRankValue : 0.000795933	name : 시민의숲역 1번출구 옆
rank : 14	pageRankValue : 0.000811767	name : 안암로터리 버스정류장 앞	rank : 14	pageRankValue : 0.000817667	name : 안암로터리 버스정류장 앞
rank : 13	pageRankValue : 0.000816233	name : 공덕역 8번 출구	rank : 13	pageRankValue : 0.000818233	name : 공덕역 8번 출구
rank : 12	pageRankValue : 0.000825133	name : 롯데월드타워 (잠실역 2번출구 쪽)	rank : 12	pageRankValue : 0.0008267	name : 롯데월드타워 (잠실역 2번출구 쪽)
rank : 11	pageRankValue : 0.0008486	name : CJ 드림시티	rank : 11	pageRankValue : 0.0008493	name : 용산역 맞은편
rank : 10	pageRankValue : 0.000859667	name : 홍대입구역 8번출구 앞	rank : 10	pageRankValue : 0.000851967	name : 구로디지털단지 지역 앞
rank : 9	pageRankValue : 0.000861067	name : 용산역 맞은편	rank : 9	pageRankValue : 0.0008604	name : 홍대입구역 8번출구 앞
rank : 8	pageRankValue : 0.000862033	name : 청계천 생태교실 앞	rank : 8	pageRankValue : 0.0008615	name : 청계천 생태교실 앞
rank : 7	pageRankValue : 0.000863733	name : 구로디지털단지 역 앞	rank : 7	pageRankValue : 0.000865133	name : CJ 드림시티
rank : 6	pageRankValue : 0.0008883	name : 봉림교 교통섬	rank : 6	pageRankValue : 0.000892267	name : 봉림교 교통섬
rank : 5	pageRankValue : 0.0009986	name : 마포구민체육센터 앞	rank : 5	pageRankValue : 0.0009819	name : 마포구민체육센터 앞
rank : 4	pageRankValue : 0.0010063	name : 둑섬유원지역 1번출구 앞	rank : 4	pageRankValue : 0.000997833	name : 둑섬유원지역 1번출구 앞
rank : 3	pageRankValue : 0.0010468	name : 옥수역 3번 출구	rank : 3	pageRankValue : 0.00104743	name : 옥수역 3번 출구
rank : 2	pageRankValue : 0.00109457	name : 고속터미널역 8-1번, 8-2번 출구 사이	rank : 2	pageRankValue : 0.00108357	name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 1	pageRankValue : 0.0011175	name : 여의나루역 1번출구 앞	rank : 1	pageRankValue : 0.00111717	name : 여의나루역 1번출구 앞

1. 원쪽:  $i = 0$ , 우측:  $i = 2,000$ 의 결과

2. 예상

A. 큰 차이가 나지 않을 것으로 예상했습니다. 시작점을 바꾸는 건데 큰 차이가..? 라고 생각했습니다.

3. 결과

A. 예상이 맞은 것 같습니다. 순위가 변수에 상관없이 대부분 큰 차이없이 나오는 것을 볼 수 있습니다.

B. 이유는 RandomWalker 가 돌아다닐 거리가 충분하면, 시작점은 크게 상관없다고 생각합니다.

## ii. n 값 실험. Rank30 위 까지만 데이터로 사용하였습니다.

rank : 30	pageRankValue : 0.005	name : 청구아파트 앞	rank : 30	pageRankValue : 0.000733543	name : 건대 입구역 사거리(롯데백화점)
rank : 29	pageRankValue : 0.005	name : 월계초인아파트 1단지	rank : 29	pageRankValue : 0.000734497	name : 반포본동 주민센터 앞
rank : 28	pageRankValue : 0.005	name : 서울과학기술대학교(어학교육원)	rank : 28	pageRankValue : 0.00073905	name : 영등포구청역 1번출구
rank : 27	pageRankValue : 0.005	name : 신천중학교 후문 맞은편 버스정류장	rank : 27	pageRankValue : 0.000743187	name : 이수역 1번출구
rank : 26	pageRankValue : 0.005	name : 가좌삼거리	rank : 26	pageRankValue : 0.00074887	name : 삼호물산버스정류장(23370) 옆
rank : 25	pageRankValue : 0.005	name : 남가좌동 제2공영주차장	rank : 25	pageRankValue : 0.000751583	name : IFC몰
rank : 24	pageRankValue : 0.005	name : 등신 병원 및 은편	rank : 24	pageRankValue : 0.000753473	name : 천호역 4번출구(현대백화점)
rank : 23	pageRankValue : 0.005	name : 광진 구청 앞	rank : 23	pageRankValue : 0.00076387	name : 응봉역 1번출구
rank : 22	pageRankValue : 0.005	name : 응봉역 1번출구	rank : 22	pageRankValue : 0.000773357	name : 마곡나루역 2번 출구
rank : 21	pageRankValue : 0.005	name : 서울숲 관리사무소	rank : 21	pageRankValue : 0.000774397	name : 밤산역 1번, 9번 인근 대여소
rank : 20	pageRankValue : 0.005	name : 광진 경찰서	rank : 20	pageRankValue : 0.00077868	name : 장안동 사거리
rank : 19	pageRankValue : 0.005	name : 광진정보도서관	rank : 19	pageRankValue : 0.00078011	name : 신도림역 1번 출구 앞
rank : 18	pageRankValue : 0.005	name : 금호나들목	rank : 18	pageRankValue : 0.00078509	name : 동방1교
rank : 17	pageRankValue : 0.005	name : 서울숲 공영주차장앞	rank : 17	pageRankValue : 0.000789297	name : 홍대입구역 3번출구
rank : 16	pageRankValue : 0.005	name : 행당동 삼부APT후문	rank : 16	pageRankValue : 0.00079079	name : 이마트 은평점
rank : 15	pageRankValue : 0.005	name : 구의7단지현대아파트	rank : 15	pageRankValue : 0.000795403	name : 시민의숲역 1번출구 옆
rank : 14	pageRankValue : 0.005	name : 나루마당	rank : 14	pageRankValue : 0.000810147	name : 안암로터리 버스정류장 앞
rank : 13	pageRankValue : 0.005	name : 건대 병원후문	rank : 13	pageRankValue : 0.000819393	name : 광덕역 8번출구
rank : 12	pageRankValue : 0.005	name : 항동지구5단지 501동 앞	rank : 12	pageRankValue : 0.000830567	name : 롯데월드타워(잠실역 2번출구 쪽)
rank : 11	pageRankValue : 0.005	name : 위경1동주민센터 앞	rank : 11	pageRankValue : 0.00083336	name : 구로디지털단지역 앞
rank : 10	pageRankValue : 0.005	name : 광흥장역 5번출구	rank : 10	pageRankValue : 0.000854647	name : 홍대입구역 8번출구 앞
rank : 9	pageRankValue : 0.005	name : 파크타워 104동 앞	rank : 9	pageRankValue : 0.000858547	name : CJ 드림시티
rank : 8	pageRankValue : 0.01	name : 약수역 10번출구 앞	rank : 8	pageRankValue : 0.000860423	name : 용산역 맞은편
rank : 7	pageRankValue : 0.01	name : 청계천 생태교 앞	rank : 7	pageRankValue : 0.00086486	name : 청계천 생태교 앞
rank : 6	pageRankValue : 0.01	name : 연의근린공원 건너편	rank : 6	pageRankValue : 0.000888817	name : 봉림교 교통섬
rank : 5	pageRankValue : 0.01	name : 석계역 문화광장 내 자전거 보관소	rank : 5	pageRankValue : 0.000985347	name : 마포구민체육센터 앞
rank : 4	pageRankValue : 0.01	name : 압구정 한양 3차 아파트	rank : 4	pageRankValue : 0.00100337	name : 뚝섬유일지역 1번출구 앞
rank : 3	pageRankValue : 0.01	name : 한국우편사 업종진흥원	rank : 3	pageRankValue : 0.00104074	name : 옥수역 3번출구
rank : 2	pageRankValue : 0.01	name : 강서공업고등학교 앞	rank : 2	pageRankValue : 0.00108934	name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 1	pageRankValue : 0.01	name : 대신파이낸스센터	rank : 1	pageRankValue : 0.00112382	name : 여의나루역 1번출구 앞

rank : 30	pageRankValue : 0.000726	name : IFC몰
rank : 29	pageRankValue : 0.00073	name : 신대방역 2번 출구
rank : 28	pageRankValue : 0.000731	name : 반포본동 주민센터 앞
rank : 27	pageRankValue : 0.000734	name : 당곡사거리
rank : 26	pageRankValue : 0.000737	name : 오류동역 2번출구
rank : 25	pageRankValue : 0.00074	name : 안암로터리 버스정류장 앞
rank : 24	pageRankValue : 0.000747	name : 사당역 12번출구 뒤
rank : 23	pageRankValue : 0.000749	name : 응봉역 1번출구
rank : 22	pageRankValue : 0.000758	name : 신도림역 1번 출구 앞
rank : 21	pageRankValue : 0.00076	name : 발산역 1번, 9번 인근 대여소
rank : 20	pageRankValue : 0.000764	name : 삼호물산버스정류장(23370) 옆
rank : 19	pageRankValue : 0.000765	name : 마곡나루역 2번 출구
rank : 18	pageRankValue : 0.000787	name : 장안동 사거리
rank : 17	pageRankValue : 0.000791	name : 동방1교
rank : 16	pageRankValue : 0.000793	name : 롯데월드타워(잠실역 2번출구 쪽)
rank : 15	pageRankValue : 0.000798	name : CJ 드림시티
rank : 14	pageRankValue : 0.000801	name : 홍대입구역 3번출구
rank : 13	pageRankValue : 0.000813	name : 홍대입구역 8번출구 앞
rank : 12	pageRankValue : 0.000823	name : 이마트 은평점
rank : 11	pageRankValue : 0.000826	name : 청계천 생태교 실 앞
rank : 10	pageRankValue : 0.000847	name : 봉림교 교통섬
rank : 9	pageRankValue : 0.000847	name : 시민의숲역 1번출구 옆
rank : 8	pageRankValue : 0.000854	name : 구로디지털단지역 옆
rank : 7	pageRankValue : 0.000855	name : 용산역 맞은편
rank : 6	pageRankValue : 0.000859	name : 광덕역 8번출구
rank : 5	pageRankValue : 0.000982	name : 뚝섬유일지역 1번출구 앞
rank : 4	pageRankValue : 0.000983	name : 옥수역 3번출구
rank : 3	pageRankValue : 0.000984	name : 마포구민체육센터 앞
rank : 2	pageRankValue : 0.001105	name : 여의나루역 1번출구 앞
rank : 1	pageRankValue : 0.001114	name : 고속터미널역 8-1번, 8-2번 출구 사이

1. 왼쪽 위 :  $n = 200$ , 오른쪽 위:  $n = 300,000,000$ , 아래쪽 :  $n = 1,000,000$

### 2. 예상

A. 아무래도 값이 클수록 더 정확한 값이 나오지 않을까...

### 3. 결과

A. 예상이 맞은 것 같습니다.

B. pagerank 값이 왼쪽 위는 노드끼리 크게 구별되지 않고 0.005 아니면 0.01로 나오는 것을 볼 수 있습니다. 어떤 노드가 중요한 노드인지 제대로 나오지 않습니다. 반면 오른쪽 위는 모든 노드끼리 충분히 구별될 정도로 pagerank 가 계산된 것을 확인할 수 있습니다. 조금 더 값을 변형시켜 돌려보았을 때,  $n$  이 1,000,000 이 넘어갈 때부터 잘 구분이 되는 것으로 파악되었습니다.

C. 이유는 RandomWalker 가 돌아다닐 횟수가 늘어나면, 더 많은 노드를 탐색할 수 있고, 그 많은 노드에 대해서 중요한 노드를 생각할 수 있기 때문이라고 생각합니다. (즉  $n=200$ 은 충분하지 않았습니다...) 그래도 300,000,000 일 때는 값을 구하는 시간도 너무 오래 걸리고, 시간과 횟수에 비해서 엄청 유의미한 결과를 보여주지는 않는 것 같아 제가 생각하는 최적은 기본 실험 수치인 30,000,000 이 적당하다고 생각합니다.

## 4. Task1 실험: 최소 어느정도 수치가 되어야 시작점에 따른 차이가 거의 없어질까요?

### A. (가중치 있는 경우)

```

try : 1
rank : 10 pageRankValue : 0.0012 name : 장승 배기 역 5번 출구
rank : 9 pageRankValue : 0.0012 name : 대로포스 터트 라운드
rank : 8 pageRankValue : 0.0013 name : 마포역 4번 출구 뒤
rank : 7 pageRankValue : 0.0013 name : 청계천 생태교 앞
rank : 6 pageRankValue : 0.0013 name : 스타벅스 앞
rank : 5 pageRankValue : 0.0013 name : 광명역 3번 출구
rank : 4 pageRankValue : 0.0014 name : 송천동 주민센터
rank : 3 pageRankValue : 0.0014 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 2 pageRankValue : 0.0015 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 1 pageRankValue : 0.0016 name : 오류동 역 2번 출구

try : 1
rank : 10 pageRankValue : 0.00089 name : IFC몰
rank : 9 pageRankValue : 0.00089 name : 옥수역 3번 출구
rank : 8 pageRankValue : 0.0009 name : 오목교 역 7번 출구 앞
rank : 7 pageRankValue : 0.00091 name : 롯데월드타워(점심역 2번 출구 쪽)
rank : 6 pageRankValue : 0.00091 name : 봉립교 교통섬
rank : 5 pageRankValue : 0.00095 name : 광명역 8번 출구
rank : 4 pageRankValue : 0.00099 name : 마포구 민화예술센터 앞
rank : 3 pageRankValue : 0.001 name : 마포터미널역의 8-1번, 8-2번 출구 사이
rank : 2 pageRankValue : 0.00105 name : 뚝섬유원지역의 1번 출구 앞
rank : 1 pageRankValue : 0.00114 name : 어의나루역 1번 출구 앞

try : 1
rank : 10 pageRankValue : 0.00083 name : 장안동 사거리
rank : 9 pageRankValue : 0.00083 name : 시민의 숲역 1번 출구 옆
rank : 8 pageRankValue : 0.00083 name : 청계천 생태교 앞
rank : 7 pageRankValue : 0.000861 name : 봉립교 교통섬
rank : 6 pageRankValue : 0.000881 name : 용산역 맞은편
rank : 5 pageRankValue : 0.000964 name : 마포구 민재유센터 앞
rank : 4 pageRankValue : 0.00183 name : 옥수역 3번 출구
rank : 3 pageRankValue : 0.00184 name : 뚝섬유원지역 1번 출구 앞
rank : 2 pageRankValue : 0.001863 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 1 pageRankValue : 0.00119 name : 어의나루역 1번 출구 앞

try : 2
rank : 10 pageRankValue : 0.00085 name : 구로디지털단지역 앞
rank : 9 pageRankValue : 0.00085 name : 장안동 사거리
rank : 8 pageRankValue : 0.00085 name : 청계천 생태교 앞
rank : 7 pageRankValue : 0.00089 name : 용산역 맞은편
rank : 6 pageRankValue : 0.00092 name : 봉립교 교통섬
rank : 5 pageRankValue : 0.000974 name : 마포구 민재유센터 앞
rank : 4 pageRankValue : 0.000991 name : 뚝섬유원지역 8-1번, 8-2번 출구 앞
rank : 3 pageRankValue : 0.00185 name : 옥수역 3번 출구 앞
rank : 2 pageRankValue : 0.00185 name : 어의나루역 1번 출구 앞
rank : 1 pageRankValue : 0.001105 name : 고속터미널역 8-1번, 8-2번 출구 사이

try : 3
rank : 10 pageRankValue : 0.000841 name : 롯데월드타워(점심역 2번 출구 쪽)
rank : 9 pageRankValue : 0.00085 name : 구로디지털단지역 앞
rank : 8 pageRankValue : 0.00085 name : CJ 드림시티
rank : 7 pageRankValue : 0.000865 name : 용산역 맞은편
rank : 6 pageRankValue : 0.00091 name : 봉립교 교통섬
rank : 5 pageRankValue : 0.000982 name : 옥수역 3번 출구
rank : 4 pageRankValue : 0.00101 name : 마포구 민재유센터 앞
rank : 3 pageRankValue : 0.00103 name : 뚝섬유원지역 1번 출구 앞
rank : 2 pageRankValue : 0.00104 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 1 pageRankValue : 0.001103 name : 어의나루역 1번 출구 앞

try : 4
rank : 10 pageRankValue : 0.000825 name : 골목역 8번 출구
rank : 9 pageRankValue : 0.000826 name : CJ 드림시티
rank : 8 pageRankValue : 0.000871 name : 봉립교 교통섬
rank : 7 pageRankValue : 0.000874 name : 청계천 생태교 앞
rank : 6 pageRankValue : 0.000876 name : 용산역 맞은편
rank : 5 pageRankValue : 0.000947 name : 마포구 민재유센터 앞
rank : 4 pageRankValue : 0.000985 name : 뚝섬유원지역 8-1번 출구 앞
rank : 3 pageRankValue : 0.00101 name : 옥수역 3번 출구
rank : 2 pageRankValue : 0.00107 name : 어의나루역 1번 출구 앞
rank : 1 pageRankValue : 0.001125 name : 고속터미널역 8-1번, 8-2번 출구 사이

try : 5
rank : 10 pageRankValue : 0.000849 name : 골목역 8번 출구
rank : 9 pageRankValue : 0.000875 name : 용산역 맞은편
rank : 8 pageRankValue : 0.000894 name : 구로디지털단지역 앞
rank : 7 pageRankValue : 0.000916 name : CJ 드림시티
rank : 6 pageRankValue : 0.000922 name : 봉립교 교통섬
rank : 5 pageRankValue : 0.000975 name : 옥수역 3번 출구 앞
rank : 4 pageRankValue : 0.000989 name : 마포구 민재유센터 앞
rank : 3 pageRankValue : 0.001026 name : 뚝섬유원지역 1번 출구 앞
rank : 2 pageRankValue : 0.001106 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 1 pageRankValue : 0.001139 name : 어의나루역 1번 출구 앞

```

- 왼쪽 :  $n = 10,000$ , 중앙 :  $n = 100,000$ , 오른쪽 :  $n = 1,000,000$
- 매 시도마다 시작점을 다르게 하였습니다.
- $n = 10,000$  일 때는 매 시도마다 pagerank 순위가 달라지는 것을 볼 수 있습니다.
- $n = 100,000$  일 때는 10,000 때 보다는 덜 합니다. 상위 rank에 대해서는 변화가 덜 하지만, 위 rank에 대해서는 매 시도마다 pagerank 순위가 꽤 달라지는 것을 볼 수 있습니다.
- $n = 1,000,000$  일 때 부터는 매 시도마다 pagerank 순위가 그래도 꽤 안정적인 모습을 볼 수 있습니다.
- 가중치가 있는 경우에서 최소  $n$ 이 최소 1,000,000 정도는 되어야 pagerank 값들이 서로 구분이 되고 시작점에 따른 차이가 거의 없어진다고 생각합니다.

## B. (가중치 없는 경우)

```

try : 1
rank : 10 pageRankValue : 0.0012 name : 보라매 역 8번 출구
rank : 9 pageRankValue : 0.0012 name : 마포구 나루역 1번 출구
rank : 8 pageRankValue : 0.0012 name : 동교동역 1번 출구
rank : 7 pageRankValue : 0.0013 name : 어의나루역 1번 출구 암
rank : 6 pageRankValue : 0.0013 name : 양평시내동쪽 주민센터 암
rank : 5 pageRankValue : 0.0013 name : 하남은행 (창동쪽으로) 입구
rank : 4 pageRankValue : 0.0014 name : 마포구 민재 옥션 터 암
rank : 3 pageRankValue : 0.0014 name : 천왕이문 하우스 1단지 암
rank : 2 pageRankValue : 0.0014 name : 천왕이문 천왕이문 1단지 암
rank : 1 pageRankValue : 0.0017 name : 신대방역 2번 출구

try : 1
rank : 10 pageRankValue : 0.00079 name : 신도림역 1번 출구 일
rank : 9 pageRankValue : 0.00086 name : 신도림역 2번 출구 일
rank : 8 pageRankValue : 0.00086 name : 신도림역 3번 출구 일
rank : 7 pageRankValue : 0.00083 name : 신도림역 4번 출구 일
rank : 6 pageRankValue : 0.00085 name : 신도림역 5번 출구 일
rank : 5 pageRankValue : 0.00086 name : 경계천 생태교실 일
rank : 4 pageRankValue : 0.00086 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 3 pageRankValue : 0.00096 name : 어의나루역 1번 출구 암
rank : 2 pageRankValue : 0.001 name : 옥성유원지역 1번 출구 암
rank : 1 pageRankValue : 0.001 name : 옥수역 3번 출구

try : 2
rank : 10 pageRankValue : 0.0008 pageRankValue : 0.00081 name : 동교동 교통섬 (2번 출구 쪽)
rank : 9 pageRankValue : 0.00081 name : 동교동 교통섬 (1번 출구 쪽)
rank : 8 pageRankValue : 0.00082 name : 동교동 교통섬 (3번 출구 쪽)
rank : 7 pageRankValue : 0.00084 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 6 pageRankValue : 0.00085 name : 동교동 교통섬 1번 출구 암
rank : 5 pageRankValue : 0.00089 name : 경계천 생태교실 쪽
rank : 4 pageRankValue : 0.00089 name : 하신16자 아파트 119동 일
rank : 3 pageRankValue : 0.00091 name : 옥성유원지 생태교실 일
rank : 2 pageRankValue : 0.00091 name : 옥수교역 3번 출구 암
rank : 1 pageRankValue : 0.00099 name : 어의나루역 1번 출구 암

try : 3
rank : 10 pageRankValue : 0.00012 name : 롯데월드타워 (잠실역 2번 출구 쪽)
rank : 9 pageRankValue : 0.00012 name : 송파구 토성1단지 102동 암
rank : 8 pageRankValue : 0.00013 name : 마포구 강정역
rank : 7 pageRankValue : 0.00013 name : 신도림역 4번 출구 일
rank : 6 pageRankValue : 0.00013 name : 신도림동 삼거리 (북·이터널방향)
rank : 5 pageRankValue : 0.00013 name : 밤반상거리 일
rank : 4 pageRankValue : 0.00014 name : 옥수교역 7번 출구 암
rank : 3 pageRankValue : 0.00014 name : 롯데월드 1번 출구
rank : 2 pageRankValue : 0.00015 name : 옥성유원지역 1번 출구 암
rank : 1 pageRankValue : 0.00017 name : 옥수역 3번 출구

try : 4
rank : 10 pageRankValue : 0.00012 name : 외국 이 대 정문 암
rank : 9 pageRankValue : 0.00012 name : 귀농·귀촌 아파트
rank : 8 pageRankValue : 0.00012 name : 신도림역 1번 출구 일
rank : 7 pageRankValue : 0.00012 name : 신도림역 2번 출구 일
rank : 6 pageRankValue : 0.00013 name : 어의나루역 1번 출구 암
rank : 5 pageRankValue : 0.00013 name : 이아시아 이엔드타운 8차 일
rank : 4 pageRankValue : 0.00013 name : 구암역 1번 출입구 일
rank : 3 pageRankValue : 0.00013 name : 노들역 3번 출구 일
rank : 2 pageRankValue : 0.00015 name : 옥수역 3번 출구
rank : 1 pageRankValue : 0.00015 name : 광나루 안전 대행 관

try : 5
rank : 10 pageRankValue : 0.00011 name : 커먼 그라운드
rank : 9 pageRankValue : 0.00012 name : 영등포 뉴타운지하상가 2번 게이트
rank : 8 pageRankValue : 0.00012 name : 오목교역 3번 출구
rank : 7 pageRankValue : 0.00012 name : 해동공동원 (천호동)
rank : 6 pageRankValue : 0.00012 name : 신도림역 4번 출구
rank : 5 pageRankValue : 0.00012 name : 신도림교자교 일
rank : 4 pageRankValue : 0.00013 name : 산계증권학교 (단현2교사거리)
rank : 3 pageRankValue : 0.00013 name : 옥수교역 2번 출구 일
rank : 2 pageRankValue : 0.00015 name : 옥성유원지역 7번 출구 일
rank : 1 pageRankValue : 0.00016 name : 옥수교역 7번 출구 일

try : 1
rank : 10 pageRankValue : 0.00079 name : 롯데월드 타워 (잠실역 2번 출구 쪽)
rank : 9 pageRankValue : 0.00084 name : 세일강간 수선역 일
rank : 8 pageRankValue : 0.00085 name : 신도림역 119동 일
rank : 7 pageRankValue : 0.00086 name : 동점유물자역 1번 출구 일
rank : 6 pageRankValue : 0.00087 name : CJ 드림타워
rank : 5 pageRankValue : 0.00089 name : 경계천 8가지거리 부근
rank : 4 pageRankValue : 0.00099 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 3 pageRankValue : 0.00101 name : 경계천 생태교실 일
rank : 2 pageRankValue : 0.00101 name : 서빙고동 주민센터 일
rank : 1 pageRankValue : 0.00104 name : 어의나루역 1번 출구 일

try : 2
rank : 10 pageRankValue : 0.00076 name : 옥산유원지 일
rank : 9 pageRankValue : 0.000761 name : 마포구 민재 옥션 터 일
rank : 8 pageRankValue : 0.000766 name : 당산육군문
rank : 7 pageRankValue : 0.000784 name : 신도림역 1번 출구 일
rank : 6 pageRankValue : 0.000797 name : 서빙고동 주민센터 일
rank : 5 pageRankValue : 0.00089 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 4 pageRankValue : 0.000928 name : 옥성유원지역 1번 출구 일
rank : 3 pageRankValue : 0.000939 name : 경계천 생태교실 일
rank : 2 pageRankValue : 0.000974 name : 어의나루역 1번 출구 일
rank : 1 pageRankValue : 0.001016 name : 옥수역 3번 출구

try : 3
rank : 10 pageRankValue : 0.000764 name : 옥산유원지
rank : 9 pageRankValue : 0.000772 name : 신도림역 1번 출구 일
rank : 8 pageRankValue : 0.000778 name : 옥수역 3번 출구
rank : 7 pageRankValue : 0.000888 name : 서빙고동 주민센터 일
rank : 6 pageRankValue : 0.000889 name : 마포구 민재 옥션 터 일
rank : 5 pageRankValue : 0.000887 name : 경계천 생태교실 일
rank : 4 pageRankValue : 0.000924 name : 옥성유원지역 1번 출구 일
rank : 3 pageRankValue : 0.000931 name : 어의나루역 1번 출구 일
rank : 2 pageRankValue : 0.00094 name : 옥수역 3번 출구
rank : 1 pageRankValue : 0.000948 name : 고속터미널역 8-1번, 8-2번 출구 사이

try : 4
rank : 10 pageRankValue : 0.000748 name : 잠실역 8번 출구
rank : 9 pageRankValue : 0.00075 name : 잠정역 7번 출구 일
rank : 8 pageRankValue : 0.000758 name : 마포구 민재 옥션 터 일
rank : 7 pageRankValue : 0.000774 name : 한신16자 아파트 119동 일
rank : 6 pageRankValue : 0.000881 name : 신도림역 119동 일
rank : 5 pageRankValue : 0.000884 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 4 pageRankValue : 0.000893 name : 경계천 생태교실 일
rank : 3 pageRankValue : 0.00091 name : 어의나루역 1번 출구 일
rank : 2 pageRankValue : 0.00094 name : 옥성유원지역 1번 출구 일
rank : 1 pageRankValue : 0.001009 name : 옥수역 3번 출구

try : 5
rank : 10 pageRankValue : 0.000739 name : 신도림역 1번 출구 일
rank : 9 pageRankValue : 0.000753 name : 옥수역 3번 출구
rank : 8 pageRankValue : 0.000777 name : 신도림역 119동 일
rank : 7 pageRankValue : 0.000793 name : 서빙고동 주민센터 일
rank : 6 pageRankValue : 0.000794 name : 마포구 민재 옥션 터 일
rank : 5 pageRankValue : 0.000991 name : 경계천 생태교실 일
rank : 4 pageRankValue : 0.000992 name : 옥성유원지역 1번 출구 일
rank : 3 pageRankValue : 0.000993 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 2 pageRankValue : 0.000943 name : 옥수역 3번 출구
rank : 1 pageRankValue : 0.000953 name : 어의나루역 1번 출구 일

try : 6
rank : 10 pageRankValue : 0.00078 name : 마포구 민재 옥션 터 일
rank : 9 pageRankValue : 0.000788 name : 신도림역 4번 출구 일
rank : 8 pageRankValue : 0.000803 name : 당산육군문
rank : 7 pageRankValue : 0.000802 name : 신도림역 119동 일
rank : 6 pageRankValue : 0.000803 name : 서빙고동 주민센터 일
rank : 5 pageRankValue : 0.000844 name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 4 pageRankValue : 0.000912 name : 경계천 생태교실 일
rank : 3 pageRankValue : 0.000924 name : 옥성유원지역 1번 출구 일
rank : 2 pageRankValue : 0.00094 name : 옥수역 3번 출구
rank : 1 pageRankValue : 0.001015 name : 옥수역 3번 출구

```

- 원쪽 :  $n = 10,000$ , 중앙 :  $n = 100,000$ , 오른쪽 :  $n = 1,000,000$
- 매 시도마다 시작점을 다르게 하였습니다.
- 원쪽 :  $n = 10,000$ , 중앙 :  $n = 100,000$ , 오른쪽 :  $n = 1,000,000$
- 매 시도마다 시작점을 다르게 하였습니다.
- 가중치가 없는 경우에서도 가중치가 있던 경우와 마찬가지로 최소  $n$ 이 최소 1,000,000 정도는 되어야 pagerank 값들이 서로 구분이 되어 시작점에 따른 차이가 거의 없어진다고 생각합니다.

### iii. q 값 실험. Rank20 위 까지만 데이터로 사용하였습니다.

rank : 28	pageRankValue : 0.00177653	name : 이마트 베스점유율 앞	rank : 28	pageRankValue : 0.0005524	name : 전주 4번 출구(한대백화점)
rank : 19	pageRankValue : 0.0018569	name : 염동포 구청역 1번출구	rank : 19	pageRankValue : 0.0005529	name : 마포구나루역 2번 출구
rank : 18	pageRankValue : 0.00181217	name : 성수 2가 1동 공영주차장 인근	rank : 18	pageRankValue : 0.000553	name : 강안동 사거리
rank : 17	pageRankValue : 0.00185953	name : 종대입구역 8번출구 앞	rank : 17	pageRankValue : 0.000553533	name : 발산역 2번, 9번 인근 대여소
rank : 16	pageRankValue : 0.00186693	name : 마포구청 역	rank : 16	pageRankValue : 0.000557833	name : 종대입구역 9번출구
rank : 15	pageRankValue : 0.00187753	name : 간대입구역 사거리(롯데백화점)	rank : 15	pageRankValue : 0.00056267	name : 서면의술역 1번출구 앞
rank : 14	pageRankValue : 0.0018892	name : 동림교 교통성	rank : 14	pageRankValue : 0.000565133	name : 안암로터리 베스점유율 앞
rank : 13	pageRankValue : 0.0018987	name : 동명 1교	rank : 13	pageRankValue : 0.000568533	name : 경덕역 8번출구
rank : 12	pageRankValue : 0.00196477	name : 구로디지털단지 역 앞	rank : 12	pageRankValue : 0.0005692	name : CJ 드림시티
rank : 11	pageRankValue : 0.001994	name : 응봉역 1번출구	rank : 11	pageRankValue : 0.000572267	name : 롯데월드타워(점실 역 2번출구 쪽)
rank : 10	pageRankValue : 0.00192287	name : 청계천 생태교실 앞	rank : 10	pageRankValue : 0.0005724	name : 구로디지털단지 역 앞
rank : 9	pageRankValue : 0.00196463	name : 이마트 은평점	rank : 9	pageRankValue : 0.000579733	name : 종대입구역 8번출구 앞
rank : 8	pageRankValue : 0.00198883	name : 자양중 앙나들목	rank : 8	pageRankValue : 0.0005811	name : 청계천 생태교실 앞
rank : 7	pageRankValue : 0.00218393	name : CJ 드림시티	rank : 7	pageRankValue : 0.000581267	name : 용산역 맞은편
rank : 6	pageRankValue : 0.0021334	name : 롯데월드타워(점실 역 2번출구 쪽)	rank : 6	pageRankValue : 0.000589633	name : 불립교 교통성
rank : 5	pageRankValue : 0.00249413	name : 옥수역 3번출구	rank : 5	pageRankValue : 0.000611833	name : 마포구민체육센터 앞
rank : 4	pageRankValue : 0.00240757	name : 고속터미널역 8-1번, 8-2번 출구 사이	rank : 4	pageRankValue : 0.000618267	name : 평선유원지역 1번출구 앞
rank : 3	pageRankValue : 0.00259417	name : 마포구민체육센터 앞	rank : 3	pageRankValue : 0.0006392	name : 옥수역 3번출구
rank : 2	pageRankValue : 0.00271327	name : 동성유원지역 1번출구 앞	rank : 2	pageRankValue : 0.0006476	name : 여의나루역 1번출구 앞
rank : 1	pageRankValue : 0.00298653	name : 여의나루역 1번출구 앞	rank : 1	pageRankValue : 0.000649533	name : 고속터미널역 8-1번, 8-2번 출구 사이

rank : 29	pageRankValue : 0.00156563	name : 신도림역 1번 출구 뒷	rank : 18	pageRankValue : 0.00157557	name : 성수 2가 1동 공영주차장 인근
rank : 19	pageRankValue : 0.001611	name : 염동포 구청역 1번출구	rank : 19	pageRankValue : 0.001611	name : 경대입구역 사거리(롯데백화점)
rank : 18	pageRankValue : 0.00165943	name : 경대입구역 사거리(롯데백화점)	rank : 18	pageRankValue : 0.00165943	name : 마포구청 역
rank : 17	pageRankValue : 0.00168383	name : 마포구청 역	rank : 17	pageRankValue : 0.00168383	name : 자양중 앙나들목
rank : 16	pageRankValue : 0.00169723	name : 자양중 앙나들목	rank : 16	pageRankValue : 0.00169723	name : 홍대입구역 8번출구 앞
rank : 15	pageRankValue : 0.0017665	name : 홍대입구역 8번출구 앞	rank : 15	pageRankValue : 0.0017665	name : 동명 1교
rank : 14	pageRankValue : 0.00177372	name : 동명 1교	rank : 14	pageRankValue : 0.00177372	name : 불립교 교통성
rank : 13	pageRankValue : 0.00176497	name : 불립교 교통성	rank : 13	pageRankValue : 0.00176497	name : 마포구민체육센터 앞
rank : 12	pageRankValue : 0.0017772	name : 응봉역 1번출구	rank : 12	pageRankValue : 0.0017772	name : 평선유원지역 1번출구
rank : 11	pageRankValue : 0.0017874	name : 응봉역 1번출구	rank : 11	pageRankValue : 0.0017874	name : 구로디지털단지 역 앞
rank : 10	pageRankValue : 0.0017874	name : 구로디지털단지 역 앞	rank : 10	pageRankValue : 0.0017874	name : 청계천 생태교실 뒷
rank : 9	pageRankValue : 0.00188543	name : 청계천 생태교실 뒷	rank : 9	pageRankValue : 0.00188543	name : 이마트 은평점
rank : 8	pageRankValue : 0.00184147	name : 이마트 은평점	rank : 8	pageRankValue : 0.00184147	name : 8번출구
rank : 7	pageRankValue : 0.0019323	name : 8번출구	rank : 7	pageRankValue : 0.0019323	name : 롯데월드타워(점실 역 2번출구 쪽)
rank : 6	pageRankValue : 0.00199633	name : CJ 드림시티	rank : 6	pageRankValue : 0.00199633	name : 옥수역 3번출구
rank : 5	pageRankValue : 0.00228167	name : 옥수역 3번출구	rank : 5	pageRankValue : 0.00228167	name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 4	pageRankValue : 0.0023206	name : 고속터미널역 8-1번, 8-2번 출구 사이	rank : 4	pageRankValue : 0.0023206	name : 마포구민체육센터 앞
rank : 3	pageRankValue : 0.00238763	name : 마포구민체육센터 앞	rank : 3	pageRankValue : 0.00238763	name : 평선유원지역 1번출구 뒷
rank : 2	pageRankValue : 0.00244987	name : 평선유원지역 1번출구 뒷	rank : 2	pageRankValue : 0.00244987	name : 여의나루역 1번출구 뒷
rank : 1	pageRankValue : 0.00274233	name : 여의나루역 1번출구 뒷	rank : 1	pageRankValue : 0.00274233	name : 고속터미널역 8-1번, 8-2번 출구 사이

1. 왼쪽 위 :  $q = 0.1$ , 오른쪽 위:  $q = 0.9$ , 아래쪽 :  $q = 0.15$

2. 예상

A.  $q$  가 작을수록 이웃 노드만 탐색하게 될 확률이 높으니... 전체 노드를 보기 힘들지 않을까?

3. 결과

A. 예상이 맞은 것 같습니다.

B. 가장 높은 pagerank 값이 왼쪽은 0.002 후반대고 오른쪽은 0.0006 중반입니다.

C. 이유는 어떤 노드의 이웃들은, 다시 그 노드와 이어져 있을 가능성이 높다고 생각합니다. 이웃 노드로 갈 확률이 높아진다는 것은 서로를 더 많이 방문하여 서로의 pagerank 값을 높여주는 것을 의미한다고 생각합니다. 따라서  $q$  가 작을 때가  $q$  가 높을 때보다 pagerank 값이 더 높게 나오는 것 같습니다.  $q$  가 작을수록 전체 노드를 탐색하기 어렵다는 것이 맞는 것 같습니다.

#### Damping factor [edit]

The PageRank theory holds that an imaginary surfer who is randomly clicking on links will eventually stop clicking. The probability, at any step, that the person will continue is a damping factor  $d$ . Various studies have tested different damping factors, but it is generally assumed that the damping factor will be set around 0.85. [5]

D. 그럼 최적의  $q$  값은 무엇인지에 대해서 궁금증이 생겨서 위키피디아에 찾아보았습니다.  $q$  라는 변수를 pagerank 에서는 damping factor 라고 명명하고 있었습니다. Damping factor 를 해석해보면, 사용자가 어떤 페이지를 보고 있고, 더 많은 정보를 위해서 그 페이지의 링크를 계속해서 클릭해 나가다가, 결국 클릭을 중단하는 시점이 오지만, 그동안 사용자가 계속해서 페이지 내의 링크를 클릭해 나갈 확률이 0.85라고 하는 것 같습니다. 그렇다면, 저의 함수에서  $q$  와 damping  $d$  는 반대의 의미를 가지기에 아마 제가 짠 pagerank 에서 최적의  $q$  는 0.15 일것이라고 생각합니다.

E. 처음엔 전체 노드에 대해서 탐색해 pagerank 를 구하는 것이 더 좋은 것이라고 생각하였습니다. 실험을 진행해 보고, 위키피디아도 찾아보니 꼭 그런 것은 아닌 것 같았습니다. Pagerank 라는 것이 이웃들이 서로를 얼마나 참조하는지에 따라서 pagerank 의 중요도를 매기는 알고리즘이 되어야하는데,  $q$  값이 높아지면, random 하게 모든 노드의 pagerank 값을 높이게 됩니다. 그렇게 되면, Pagerank 의 설계의도와 상당히 떨어지게 된다고 생각합니다. Pagerank 는 사용자가 페이지 내의 링크를 계속 따라가며 그 페이지에 대한 우선도를 매기는 것인데,  $q$  가 높다면, 그 사용자는 거의 링크를 따라가지 않고, 계속해서 새로운 내용을 검색하는 것에 해당됩니다.  $q$  반면,  $q$  가 낮다면, 계속 연결되는 페이지를 따라가며 검색하는 것이 됩니다. 위키피디아와 앞의 내용을 바탕으로 생각해 보았을 때, 실제 pagerank 가 의도 하는 설계, 그리고 사람의 검색 패턴을 고려하면, 가장 유의미한 값을 얻기 위한  $q$  는 0.15라고 생각합니다.

B. Pagerank 가 적은 노드를 시작 노드로 선택했을 때와 Pagerank 가 큰 노드를 시작 노드로 선택했을 때와 차이 분석 (i 가 98 인 경우 id: 207, name: “여의나루역 1 번출구 앞”인 노드를 가리키게 됩니다. 이는 앞선 결과에서 2 위인 노드입니다, i 가 145 인 경우 id: 257 name: 신길삼거리라는 sort 를 역순으로 했을 때 가장 작은 pagerank 로 나타난 노드입니다. 두 노드를 기준으로 테스트하였습니다. 그리고 이번 실험은 q 가 낮은 것으로 해야 결과가 명확할 것 같아 앞의 조건에서 q 만 0.2 로 낮추었습니다.)

rank : 50	pageRankValue : 0.00121873	name : 가양역 8번 출구	rank : 50	pageRankValue : 0.0012156	name : 한성백제역 2번 출구
rank : 49	pageRankValue : 0.0012227	name : 오목교역 7번 출구 앞	rank : 49	pageRankValue : 0.0012195	name : 가양역 8번 출구
rank : 48	pageRankValue : 0.00122633	name : 국민일보 앞	rank : 48	pageRankValue : 0.0012221	name : 마곡나루역 5번 출구 뒤편
rank : 47	pageRankValue : 0.00123717	name : 북동역5번출구 교통정보센터 앞	rank : 47	pageRankValue : 0.00123163	name : 국민일보 앞
rank : 46	pageRankValue : 0.0012399	name : 한성백제역 2번 출구	rank : 46	pageRankValue : 0.00123237	name : 북동역5번출구 교통정보센터 앞
rank : 45	pageRankValue : 0.00125387	name : 근로자회관 사거리	rank : 45	pageRankValue : 0.00124613	name : 근로자회관 사거리
rank : 44	pageRankValue : 0.00126763	name : 장안동워더스빌앞	rank : 44	pageRankValue : 0.00125317	name : 장안동워더스빌앞
rank : 43	pageRankValue : 0.0012715	name : 당곡사거리	rank : 43	pageRankValue : 0.00126468	name : 당곡사거리
rank : 42	pageRankValue : 0.00127547	name : 오목교역 3번 출구	rank : 42	pageRankValue : 0.0012717 name : 오목교역 3번 출구	name : 오목교역 3번 출구
rank : 41	pageRankValue : 0.0012807	name : 방이삼거리	rank : 41	pageRankValue : 0.00127407	name : 방이삼거리
rank : 40	pageRankValue : 0.00129157	name : 센트럴 푸르지오 시티 앞	rank : 40	pageRankValue : 0.00128467	name : 센트럴 푸르지오 시티 앞
rank : 39	pageRankValue : 0.00130433	name : 노원역 1번 출구	rank : 39	pageRankValue : 0.0013052	name : 노원역 1번 출구
rank : 38	pageRankValue : 0.00132613	name : 시민의 숲역 1번 출구 옆	rank : 38	pageRankValue : 0.0013087	name : 시민의 숲역 1번 출구 옆
rank : 37	pageRankValue : 0.00133043	name : 이수역 1번 출구	rank : 37	pageRankValue : 0.00132197	name : 신대방역 2번 출구
rank : 36	pageRankValue : 0.00133393	name : 신대방역 2번 출구	rank : 36	pageRankValue : 0.00132997	name : 이수역 1번 출구
rank : 35	pageRankValue : 0.0013418	name : 영등포역 하소평센타 5번 출구	rank : 35	pageRankValue : 0.00134447	name : 영등포역 하소평센타 5번 출구
rank : 34	pageRankValue : 0.00134693	name : 천호역 4번 출구 (현대백화점)	rank : 34	pageRankValue : 0.0013484	name : 천호역 4번 출구 (현대백화점)
rank : 33	pageRankValue : 0.00135917	name : 둔촌역 3번 출입구	rank : 33	pageRankValue : 0.00135297	name : 둔촌역 3번 출입구
rank : 32	pageRankValue : 0.0013803	name : 궁동역 1번 출구 앞	rank : 32	pageRankValue : 0.00139837	name : 망원역 1번 출구 앞
rank : 31	pageRankValue : 0.00139353	name : 마곡나루역 2번 출구	rank : 31	pageRankValue : 0.00139147	name : 마곡나루역 2번 출구
rank : 30	pageRankValue : 0.0013998	name : 망원역 1번 출구 앞	rank : 30	pageRankValue : 0.00140433	name : 장안동 사거리
rank : 29	pageRankValue : 0.00140237	name : 장안동 사거리	rank : 29	pageRankValue : 0.00140493	name : 공릉역 1번 출구 앞
rank : 28	pageRankValue : 0.00143183	name : 이마트 버스정류소 옆	rank : 28	pageRankValue : 0.00142553	name : 이마트 버스정류소 옆
rank : 27	pageRankValue : 0.00143653	name : 성수 2가1동 공영주차장 인근	rank : 27	pageRankValue : 0.00143623	name : 성수 2가1동 공영주차장 인근
rank : 26	pageRankValue : 0.0014392	name : 홍대입구역 3번 출구	rank : 26	pageRankValue : 0.00143793	name : 발산역 1번, 9번 인근 대여소
rank : 25	pageRankValue : 0.00144023	name : 발산역 1번, 9번 인근 대여소	rank : 25	pageRankValue : 0.00144402	name : IFC몰
rank : 24	pageRankValue : 0.00144103	name : IFC몰	rank : 24	pageRankValue : 0.00144428	name : 홍대입구역 3번 출구
rank : 23	pageRankValue : 0.00146007	name : 잠실역 8번 출구	rank : 23	pageRankValue : 0.0014511	name : 잠실역 8번 출구
rank : 22	pageRankValue : 0.0014722	name : 용산역 맞은편	rank : 22	pageRankValue : 0.0014708	name : 용산역 맞은편
rank : 21	pageRankValue : 0.0014734	name : 신도림역 1번 출구 앞	rank : 21	pageRankValue : 0.0014791	name : 신도림역 1번 출구 앞
rank : 20	pageRankValue : 0.00148483	name : 안암로터리 버스정류장 앞	rank : 20	pageRankValue : 0.0014934	name : 안암로터리 버스정류장 앞
rank : 19	pageRankValue : 0.00149597	name : 영등포구청역 1번 출구	rank : 19	pageRankValue : 0.00149927	name : 영등포구청역 1번 출구
rank : 18	pageRankValue : 0.00149943	name : 공덕역 8번 출구	rank : 18	pageRankValue : 0.00150843	name : 공덕역 8번 출구
rank : 17	pageRankValue : 0.00152567	name : 자양중앙나들목	rank : 17	pageRankValue : 0.00151357	name : 자양중앙나들목
rank : 16	pageRankValue : 0.00152757	name : 건대입구역 사거리 (롯데백화점)	rank : 16	pageRankValue : 0.001521	name : 건대입구역 사거리 (롯데백화점)
rank : 15	pageRankValue : 0.00155437	name : 마포구청역	rank : 15	pageRankValue : 0.00156583	name : 마포구청역
rank : 14	pageRankValue : 0.00163373	name : 홍대입구역 8번 출구 앞	rank : 14	pageRankValue : 0.00163683	name : 홍대입구역 8번 출구 앞
rank : 13	pageRankValue : 0.00164277	name : 동방1교	rank : 13	pageRankValue : 0.00165147	name : 동방1교
rank : 12	pageRankValue : 0.00164533	name : 응봉역 1번 출구	rank : 12	pageRankValue : 0.0016554	name : 응봉역 1번 출구
rank : 11	pageRankValue : 0.00168837	name : 구로디지털단지역 앞	rank : 11	pageRankValue : 0.00168267	name : 구로디지털단지역 앞
rank : 10	pageRankValue : 0.00169117	name : 불림교 교통섬	rank : 10	pageRankValue : 0.00169713	name : 불림교 교통섬
rank : 9	pageRankValue : 0.00172857	name : 청계천 생태교실 앞	rank : 9	pageRankValue : 0.00171333	name : 청계천 생태교실 앞
rank : 8	pageRankValue : 0.0017304	name : 이마트 은평점	rank : 8	pageRankValue : 0.00173097	name : 이마트 은평점
rank : 7	pageRankValue : 0.0018074	name : 롯데월드타워 (잠실역 2번 출구 쪽)	rank : 7	pageRankValue : 0.0018008	name : 롯데월드타워 (잠실역 2번 출구 쪽)
rank : 6	pageRankValue : 0.00183676	name : CJ 드림시티	rank : 6	pageRankValue : 0.00186127	name : CJ 드림시티
rank : 5	pageRankValue : 0.0021527	name : 옥수역 3번 출구	rank : 5	pageRankValue : 0.00214727	name : 옥수역 3번 출구
rank : 4	pageRankValue : 0.0022191	name : 고속터미널역 8-1번, 8-2번 출구 사이	rank : 4	pageRankValue : 0.0022255	name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 3	pageRankValue : 0.0022341	name : 마포구민체육센터 앞	rank : 3	pageRankValue : 0.0022385	name : 마포구민체육센터 앞
rank : 2	pageRankValue : 0.00229687	name : 똑섬유월지역 1번 출구 앞	rank : 2	pageRankValue : 0.00229493	name : 똑섬유월지역 1번 출구 앞
rank : 1	pageRankValue : 0.00260303	name : 여의나루역 1번 출구 앞	rank : 1	pageRankValue : 0.00259307	name : 여의나루역 1번 출구 앞

1. 왼쪽: i = 98, 우측: i = 145 의 결과

2. 예상

A. 시도 횟수가 정말 많아서 결과는 전체적으로 비슷하게 나올거 같은데.. 그래도 시작 노드를 애초부터 rank 값이 높은 노드로 하면.. 그래도 값이 조금 더 높지 않을까..?

3. 결과

A. 예상이...맞다고 보기엔 민망하다고 생각합니다.

B. 아주 미미한 차이이지만, 0.0001 차이가 발생하는 것 같습니다. 음.. 차이는 발생하지만, pagerank 결과에 영향을 미치는 유의미한 결과라고 보기는 어려울 것 같습니다. 시도 횟수가 많으면 이 실험에서 유의미한 차이가 발생하지 않을 것 같습니다.

C. 방향성이 없는 그래프를 방향성이 있는 그래프로 만들고, 방향성이 있는 그래프를 방향성이 있는 그래프로 만들었을 때의 결과는 어떨까? 앞서 있었던 실험의 결과를 토대로 제일 적당한 값이라고 생각되는  $I = 0$ ,  $n = 30,000,000$ ,  $q = 0.15$  값을 토대로 실험하였습니다. 가중치가 있는 그래프로 실험할 경우, 서로를 가르키는 가중치가 다를 경우에 대해서 처리 방법이 마땅히 떠오르지 않고, 전체 데이터를 활용하기 힘들 것 같아 가중치가 없는 그래프를 사용하였습니다.

#### i. 방향성이 없는 그래프를 방향성이 있는 그래프로 생성. STARWARS 데이터를 사용.

rank : 30	pageRankValue : 0.00927963	name : CAPTAIN PANAKA	rank : 30	pageRankValue : 0.0111733	name : HAN
rank : 29	pageRankValue : 0.00931123	name : KYLO REN	rank : 29	pageRankValue : 0.0117111	name : TEY HOW
rank : 28	pageRankValue : 0.00939367	name : MON MOTHMA	rank : 28	pageRankValue : 0.011771	name : SEBULBA
rank : 27	pageRankValue : 0.0097805	name : ADMIRAL ACKBAR	rank : 27	pageRankValue : 0.0118361	name : TARKIN
rank : 26	pageRankValue : 0.0105927	name : KI-ADI-MUNDI	rank : 26	pageRankValue : 0.0118966	name : JAR JAR
rank : 25	pageRankValue : 0.0107184	name : LANDO	rank : 25	pageRankValue : 0.0134007	name : OWEN
rank : 24	pageRankValue : 0.0108994	name : WEDGE	rank : 24	pageRankValue : 0.0142093	name : SENATOR ASK AAK
rank : 23	pageRankValue : 0.0114496	name : BOBA FETT	rank : 23	pageRankValue : 0.0144781	name : MON MOTHMA
rank : 22	pageRankValue : 0.0122922	name : FINN	rank : 22	pageRankValue : 0.0151531	name : PIETT
rank : 21	pageRankValue : 0.0131236	name : JABBA	rank : 21	pageRankValue : 0.0155253	name : NUTE GUNRAY
rank : 20	pageRankValue : 0.0131784	name : GENERAL HUX	rank : 20	pageRankValue : 0.0157266	name : SIO BIBBLE
rank : 19	pageRankValue : 0.0143283	name : MACE WINDU	rank : 19	pageRankValue : 0.0158797	name : REY
rank : 18	pageRankValue : 0.0146809	name : BB-8	rank : 18	pageRankValue : 0.0167999	name : TAUN WE
rank : 17	pageRankValue : 0.0164287	name : POE	rank : 17	pageRankValue : 0.0175032	name : LEIA
rank : 16	pageRankValue : 0.017075	name : BAIL ORGANA	rank : 16	pageRankValue : 0.0180125	name : SOLA
rank : 15	pageRankValue : 0.0176557	name : YODA	rank : 15	pageRankValue : 0.0184173	name : RIC OLIE
rank : 14	pageRankValue : 0.0191041	name : LEIA	rank : 14	pageRankValue : 0.019108	name : TC-14
rank : 13	pageRankValue : 0.0195185	name : NUTE GUNRAY	rank : 13	pageRankValue : 0.0199367	name : UNKAR PLUTT
rank : 12	pageRankValue : 0.0195822	name : CHEWBACCA	rank : 12	pageRankValue : 0.0243775	name : SUN RIT
rank : 11	pageRankValue : 0.0218033	name : LUKE	rank : 11	pageRankValue : 0.0246728	name : POE
rank : 10	pageRankValue : 0.0235041	name : DARTH VADER	rank : 10	pageRankValue : 0.0282343	name : SHMI
rank : 9	pageRankValue : 0.0235308	name : R2-D2	rank : 9	pageRankValue : 0.0298786	name : LUKE
rank : 8	pageRankValue : 0.0243798	name : JAR JAR	rank : 8	pageRankValue : 0.0299195	name : WEDGE
rank : 7	pageRankValue : 0.0253649	name : EMPEROR	rank : 7	pageRankValue : 0.0300091	name : OBI-WAN
rank : 6	pageRankValue : 0.0259109	name : HAN	rank : 6	pageRankValue : 0.0304663	name : QUI-GON
rank : 5	pageRankValue : 0.0298828	name : QUI-GON	rank : 5	pageRankValue : 0.0357547	name : SNAP
rank : 4	pageRankValue : 0.0327981	name : C-3PO	rank : 4	pageRankValue : 0.0445031	name : PADME
rank : 3	pageRankValue : 0.0330795	name : PADME	rank : 3	pageRankValue : 0.0479486	name : YODA
rank : 2	pageRankValue : 0.0415683	name : ANAKIN	rank : 2	pageRankValue : 0.0523612	name : R2-D2
rank : 1	pageRankValue : 0.0417291	name : OBI-WAN	rank : 1	pageRankValue : 0.0590216	name : WATTO

1. 원쪽: 방향성이 없도록, 오른쪽: 방향성이 있도록

2. 예상

- A. 스타워즈 데이터 같은 경우에 데이터 자체 크기가 작아서 서로를 가르키는 경우가 거의 없는 것 같다. 페이지랭크를 돌리면, 링크드리스트처럼 연결되어 예상하는 결과와 다를 것 같다.

3. 결과

- A. 스타워즈 데이터는 양방향 처리를 해주어야 하는데 그렇지 않다보니 이전 결과와 많은 차이가 나는 것을 알 수 있다.

## ii. 방향성이 있는 그래프를 방향성이 없는 그래프로 생성. bicycle 데이터를 사용.

rank : 30	pageRankValue : 0.0010137	name : 옥포교역 3번출구	rank : 30	pageRankValue : 0.00107183	name : 용산역 맞은편
rank : 29	pageRankValue : 0.00101533	name : 용산역 맞은편	rank : 29	pageRankValue : 0.0010748	name : 용두역 4번출구
rank : 28	pageRankValue : 0.00101533	name : 잠실나들목2(엘스아파트136동)	rank : 28	pageRankValue : 0.0010806	name : 홍대입구역 8번출구 앞
rank : 27	pageRankValue : 0.00102073	name : 센트럴 푸르지오 시티 앞	rank : 27	pageRankValue : 0.00108173	name : 잠실나들목2(엘스아파트136동)
rank : 26	pageRankValue : 0.00102733	name : 동묘앞역 1번출구 뒤	rank : 26	pageRankValue : 0.00108273	name : 양평과나들목 보행통로 입구
rank : 25	pageRankValue : 0.0010357	name : 구로디지털단지역 앞	rank : 25	pageRankValue : 0.00108373	name : 배곧가로앞
rank : 24	pageRankValue : 0.00103777	name : 공덕역 8번출구	rank : 24	pageRankValue : 0.00108477	name : 잠실역 6번출구
rank : 23	pageRankValue : 0.00104337	name : 봉래교 교통섬	rank : 23	pageRankValue : 0.001095	name : 센트럴 푸르지오 시티 앞
rank : 22	pageRankValue : 0.0010478	name : 광화문역 5번출구	rank : 22	pageRankValue : 0.00110253	name : 마포역 2번출구 뒤
rank : 21	pageRankValue : 0.0010784	name : 흑석역 1번출구	rank : 21	pageRankValue : 0.00112063	name : 오룡교역 3번출구
rank : 20	pageRankValue : 0.0010919	name : 청계8가사거리 부근	rank : 20	pageRankValue : 0.001136	name : 반포쇼핑타운 2동 앞
rank : 19	pageRankValue : 0.00110357	name : 홍대입구역 8번출구 앞	rank : 19	pageRankValue : 0.00114533	name : 잠실나루역 (2번 출구 쪽)
rank : 18	pageRankValue : 0.0011062	name : 마포역 2번출구 뒤	rank : 18	pageRankValue : 0.00114677	name : 영등포역지하쇼핑센타 5번출구
rank : 17	pageRankValue : 0.00111347	name : 영등포역지하쇼핑센타 5번출구	rank : 17	pageRankValue : 0.00115623	name : 흑석역 1번출구
rank : 16	pageRankValue : 0.0011221	name : 현대고등학교 건너편	rank : 16	pageRankValue : 0.00122116	name : 현대고등학교 건너편
rank : 15	pageRankValue : 0.00117047	name : 한신16차아파트 119동 앞	rank : 15	pageRankValue : 0.0013287	name : 한신16차아파트 119동 앞
rank : 14	pageRankValue : 0.0011868	name : 롯데월드타워(잠실역2번출구 쪽)	rank : 14	pageRankValue : 0.00135207	name : 롯데월드타워(잠실역2번출구 쪽)
rank : 13	pageRankValue : 0.0012663	name : 잠실역 8번출구	rank : 13	pageRankValue : 0.0013614	name : IFC몰
rank : 12	pageRankValue : 0.00125837	name : 마포구민체육센터 앞	rank : 12	pageRankValue : 0.00136167	name : 잠실역 8번출구
rank : 11	pageRankValue : 0.00126027	name : 서빙고동 주민센터 앞	rank : 11	pageRankValue : 0.00141617	name : 신도림역 1번 출구 앞
rank : 10	pageRankValue : 0.00130937	name : 신도림역 1번 출구 앞	rank : 10	pageRankValue : 0.00143293	name : 마포구민체육센터 앞
rank : 9	pageRankValue : 0.0013198	name : IFC몰	rank : 9	pageRankValue : 0.0014485	name : 합정역 7번출구 앞
rank : 8	pageRankValue : 0.0013635	name : 당산육감문	rank : 8	pageRankValue : 0.00146763	name : 응봉역 1번출구
rank : 7	pageRankValue : 0.00137227	name : 응봉역 1번출구	rank : 7	pageRankValue : 0.001484	name : 서빙고동 주민센터 앞
rank : 6	pageRankValue : 0.0014306	name : 학정역 7번출구 앞	rank : 6	pageRankValue : 0.00157473	name : 당산육감문
rank : 5	pageRankValue : 0.00161913	name : 고속터미널역 8-1번, 8-2번 출구 사이	rank : 5	pageRankValue : 0.0017493	name : 고속터미널역 8-1번, 8-2번 출구 사이
rank : 4	pageRankValue : 0.00174853	name : 뚝섬유원지역 1번출구 앞	rank : 4	pageRankValue : 0.0018021	name : 청계천 생태교실 앞
rank : 3	pageRankValue : 0.00176743	name : 청계천 생태교실 앞	rank : 3	pageRankValue : 0.0020185	name : 옥수역 3번출구
rank : 2	pageRankValue : 0.00185203	name : 옥수역 2번출구	rank : 2	pageRankValue : 0.00204627	name : 뚝섬유원지역 1번출구 앞
rank : 1	pageRankValue : 0.00186823	name : 여의나루역 1번출구 앞	rank : 1	pageRankValue : 0.0020505	name : 여의나루역 1번출구 앞

1. 왼쪽: 방향성이 없도록, 오른쪽: 방향성이 있도록

### 2. 예상

A. 따릉이 데이터는 스타워즈 데이터와 다르게 데이터 양이 방대하고, 스타워즈 데이터는 서로를 가리키는 경우가 데이터에 잘 없지만, 따릉이 데이터에는 무수한 데이터 속에 있을지도 모르므로, 결과가 조금 다르더라도, 엄청나게 다르지는 않을 것 같다.

### 3. 결과

A. 1~30 위에 있는 노드들은 크게 달라지지 않았지만, 노드들의 순서가 많이 변한 것을 볼 수 있다. 보다 정확한 결과는 방향성이 있도록 한 것이 더 정확한 것 같다.

## iii. 총 결과

1. 데이터의 특성을 잘 파악하고 그에 맞는 데이터 분석법을 채택하는 것이 가장 좋을 것 같다. 스타워즈 데이터는 영화에 한 장면에 나오는 인물을 파악한 것으로, 뚜렷한 방향성이 없는 데이터 특성이므로, 그에 따라 방향성이 없는 그래프를 생성하여 분석하는 것이 가장 합리적이고, 따릉이 데이터는 출발지와 목적지가 있는 뚜렷한 방향성이 있는 데이터 특성이므로 방향성이 있는 그래프를 생성하여 분석하는 것이 가장 합리적인 것으로 판단됩니다.