**공간 데이터 관리 및 응용 Term Project document**

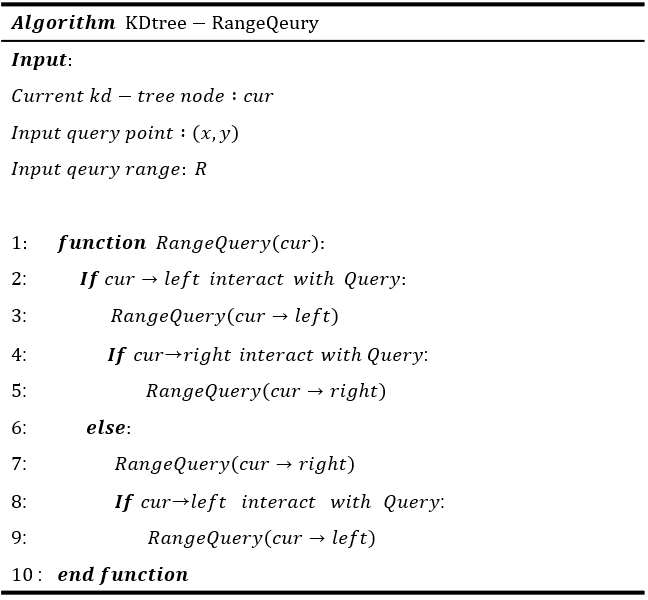
120210198 윤동성

1. **프로젝트 목표**

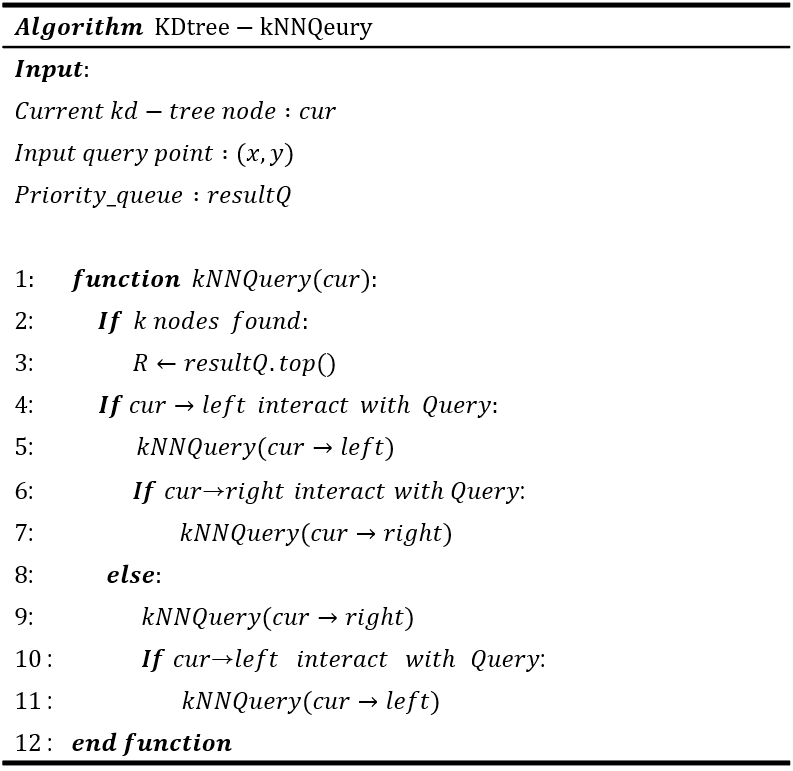
공간 인덱스인 kd-tree, R-tree를 활용하여 range query 및 kNN query processing algorithm을 설계하고 질의 처리 성능을 비교 분석한다. 구체적으로는 다음 세 가지 방법을 사용하여 질의를 처리하고 질의 처리에 걸리는 시간, 질의 처리 과정에서 검사되는 객체의 수를 비교 측정한다.  
1. Brute-force 알고리즘: 전 탐색을 통해 질의 조건에 부합하는 객체를 검색한다.

1. kd-tree: kd-tree 인덱스를 생성하고 DFS, 혹은 BFS(best-first search) 탐색을 통해 질의 조건에 부합하는 객체를 검색한다.
2. R-tree: R-tree 인덱스를 생성하고 DFS, 혹은 BFS(best-first search) 탐색을 통해 질의 조건에 부합하는 객체를 검색한다.
3. **pseudo code**

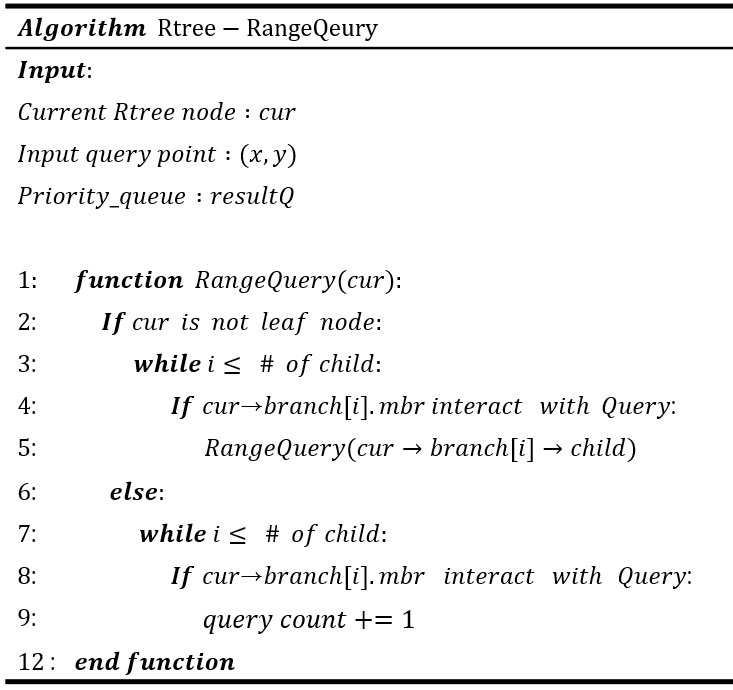
* **KDtree - rangeQeury**



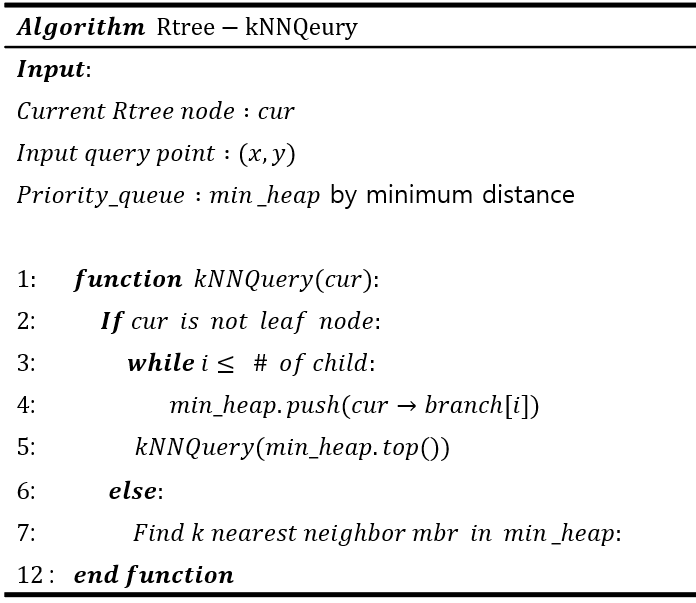
* **KDtree - kNNQeury**



* **Rtree – RangeQuery**



* **Rtree – kNNQuery**



1. **성능 분석**

* **실행 환경의 경우 visual studio 2007을 이용했다.**

**Clustered Dataset에 대한 성능비교**

**Gaussian Dataset에 대한 성능비교**

**Uniformed Dataset에 대한 성능비교**

전반적으로 kdtree-rtree-bf 순으로 성능 차이가 나타났음을 확인할 수 있었다. 다만, 내가 생각하기로는 분명 계산 횟수가 다르기에, 쿼리가 많아질수록 KD트리가 더 시간이 많이 걸리는 것으로 역전하는 부분이 나올 줄 알았으나, 이 부분이 드러나지 않았다.

이는 아쉬운 부분으로써 코드의 출력 방식에 따라서도 시간이 상당히 달라지는 것으로 보아, 데이터 개수가 적고, 비교군이 적은 것이 요인으로 보인다.