

연결리스트(LinkedList)

Q. 새로운 것을 삽입, 삭제하는 트랜잭션의 경우 Array와 Linked List중 어떤 것의 성능이 더 좋을까요? 배열 각각의 특징을 기반으로 설명해주세요

A. **Linked List**는 레퍼런스를 기반으로 삽입, 삭제 연산을 진행하기 때문에 자료의 최대 개수에 관계없이 각각 $O(1)$ 개의 레퍼런스만을 갱신하므로 일반적인 경우 $O(1)$ 의 시간이 소요됨. 반면 **Array**는 데이터의 삽입 또는 삭제 시 list의 크기를 삽입 또는 삭제할 데이터의 개수만큼 더하거나 빼고 인덱스를 조정하는 과정을 거쳐야 하기 때문에 $O(N)$ 의 시간복잡도가 소요된다. 때문에 **LinkedList**가 시간복잡도 측면에서 성능이 더 좋다

*하지만 linked list도 탐색을 통한 삽입, 삭제를 해야 할 경우 $O(N)$ 의 시간 복잡도가 걸리기도 함

자료구조: Linked List 대 ArrayList

2014년 모두들 어떤 목적과 계획을 갖고 살고 계신지요? 저는 올 한해 "Go to the Base"를 목표로 여러 계획을 세웠는데요. 그 중 하나가 과거 5년 동안 저를 되 돌아 보고 부족했던 기본 지식을 탄탄히 하는 것

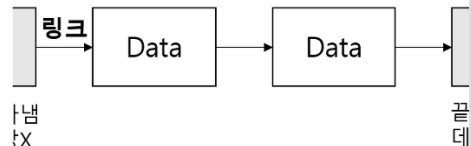
📄 <https://www.nextree.co.kr/p6506/>



[자료구조] Array VS LinkedList

Array 배열의 크기는 한 번 정하면 변경할 수 없다. 배열 초기화 시에 메모리에 할당되어 ArrayList보다 속도가 빠르다. 논리적 저장 순서와 물리적 저장 순서가 일치한다. 인덱스를 사용해 특정 원소에 접근이 가

📄 <https://hee96-story.tistory.com/46>




Q. 단순연결리스트, 이중연결리스트, 원형 연결리스트를 연결리스트의 기본 구조 관점에서 비교하세요

A. 단순 연결 리스트는 노드에 다음 노드의 주소를 가리키는 정보만 추가되어있는 가장 단순한 형태의 연결 리스트입니다. 이중 연결리스트는 다음 노드의 참조뿐만 아니라 이전 노드의 참조도 같이 가리키게한 리스트 자료구조입니다. 마지막으로 단순 연결 리스트에서 마지막 원소를 null이 아닌 첫 노드의 주소를 가리키게하면 원형 연결 리스트 자료구조가 됩니다.

*추후 참고 링크

Linked List |  Tech Interview

Ready for Tech-Interview

 <https://gyoogle.dev/blog/computer-science/data-structure/Linked%20List.html>