● Array와 Linked List를 비교해서 설명해라
▶ Array와 Linked List의 주된 차이점들은 메모리 구조에 기인합니다. Array는 메모리 상에서 연속적으로 데이터를 저장하는 자료구조 입니다. 반면, Linked List는 메모리상에서는 연속적이지 않지만, 각각의 노드가 다음 원소의 메모리 주소값을 저장해 놓음으로써 논리적 연속성을 유지합니다. 이런 메모리 구조의 차이로 인해 operation의 시간복잡도가 다릅니다. 데이터 조회는 Array는 O(1), Linked list는 O(n)의 시간복잡도를 갖습니다. 반면 삽입/삭제는 Array는 O(n), Linked list는 O(1)의 시간복잡도를 갖습니다. 따라서 얼마만큼의 데이터를 저장할지 미리 알고 있고, 조회를 많이 한다면 Array를 사용하는 것이 좋습니다. 반면에 몇개의 데이터를 저장할지 불확실하고, 삽입 삭제가 잦다면 Linked list를 사용하는 것이 유리합니다. 상황에 따라 메모리를 효율적으로 사용할 수 있는 자료구조가 달라지기 때문에, 자료구조를 상황에 따라 적절하게 선택하는 것이 중요합니다.

▶ **조회 (lookup):** Array는 메모리상에서 연속적으로 데이터를 저장하였기 때문에 저장된 데이터에 즉시 접근(random access)할 수 있어 조회에 O(1)의 시간이 걸립니다. 이와 반면 Linked List는 메모리 상에서 불연속적으로 데이터를 저장하기 때문에 순차 접근(Sequential Access)만 가능합니다. 그렇기 때문에, 특정 index의 데이터를 조회하기 위해서는 O(n)의 시간이 걸리게 됩니다.

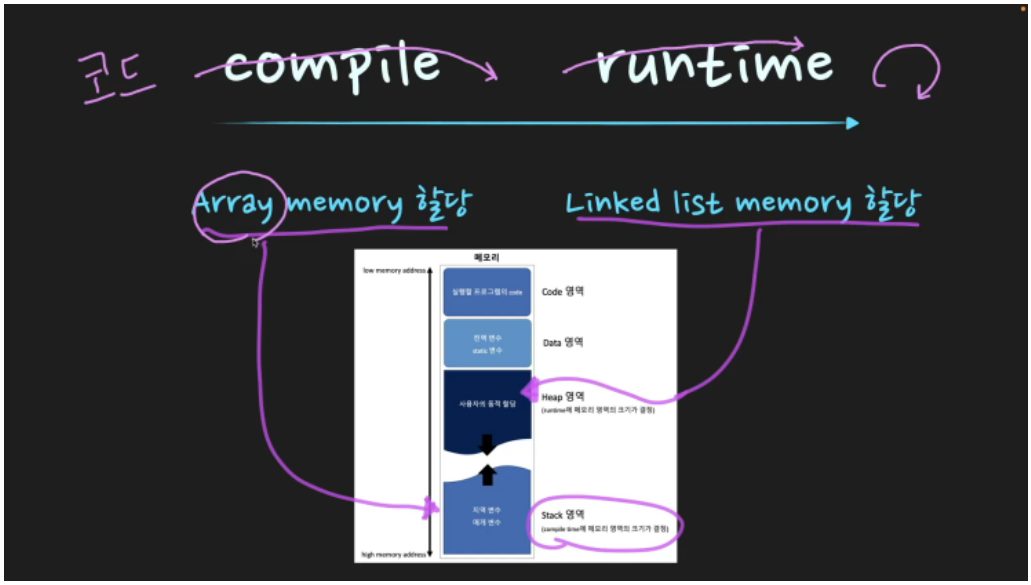
▶**삽입/삭제 (insert/delete):** Array의 경우 맨 마지막 원소를 추가/삭제하면 시간복잡도가 O(1)입니다. 하지만 맨 마지막 원소가 아닌 중간에 있는 원소를 삽입/삭제하면 그 원소보다 큰 인덱스의 원소들을 한 칸씩 shift 해줘야 하는 비용(cost)이 발생하여, 시간복잡도가 O(n)이 됩니다. 반면, Linked List는 어느 원소를 추가/삭제 하더라도 node에서 다음주소를 가르키는 부분만 다든 주소 값으로 변경하면 되기 때문에 shift할 필요 없어서 시간복잡도가 O(1)입니다. 하지만 Linked list의 경우 추가/삭제를 하려는 index의 원소까지 도달하는데 O(n)의 시간이 걸리기 때문에, 실질적으로 Linked List도 추가/삭제 시에 O(n)의 시간이 걸린다고 볼 수 있습니다.

▶**memory:** Array의 주된 장점은 데이터 접근과 append(배열 마지막 끝에 원소 추가)가 빠르다는 것입니다. 하지만 메모리 낭비라는 단점이 있습니다. 배열은 선언시에 고정된 크기를 설정하여 메모리 할당을 하기 때문에, 데이터가 저장되어 있지 않더라도 메모리를 차지하여 메모리 낭비가 발생합니다. 이와 반면 Linked List는 런타임 중에서도 size를 늘리고 줄일 수 있습니다. 그래서 initial size를 고민할 필요가 없고, 필요한 만큼만 메모리를 할당하여 메모리 낭비가 없습니다. 하지만 Linked List는 각 노드마다 값만 저장하는 것이 아니라 그 다음 노드의 주소까지 저장해야 해서 Array보다 더 많은 메모리를 차지하기 때문에, 미리 들어올 데이터의 양을 알고만 있다면 Array가 메모리를 더 효율적으로 사용합니다.

● 어느 상황에 Linked list를 쓰는게 Array보다 더 나을까?
▶ 얼마만큼의 데이터가 들어올지 예측을 할 수 없을 때, O(1)으로 삽입/삭제를 자주 해야 될 때, 그리고 조회 작업을 별로 하지 않을 때 입니다.

● 어느 상황에 Array를 쓰는게 Linked list보다 더 나을까?
▶ 선언 당시에 데이터의 갯수를 미리 알고 있을때, 조회 작업을 자주 해야될 때, 데이터를 반복문을 통해서 빠르게 순회할 때, 그리고 메모리를 적게 쓰는게 중요한 상황일 때 입니다. Linked list보단 Array가 메모리를 적게 차지 하기 때문에 미리 들어올 데이터의 양을 알고만 있다면 Array가 메모리를 더 효율적으로 사용합니다.

● Array와 Linked List의 memory allocation은 언제 일어나며, 메모리의 어느 영역을 할당 받나?
▶ Array는 컴파일 단계에서 메모리 할당이 일어나고, 이를 정적 메모리 할당(Static Memory Allocation)이라고 합니다. 이 경우, Stack 메모리 영역에 할당됩니다. Linked List의 경우는 런타임 단계에서 새로운 노드가 추가될 때마다 메모리 할당이 일어납니다. 이를 동적 메모리 할당(Dynamic Memory Allocation)이라고 하고, 이 경우 Heap 메모리 영역에 할당됩니다.



Resources: inflearn 개발남 노씨