

알고리즘 스터디 16회

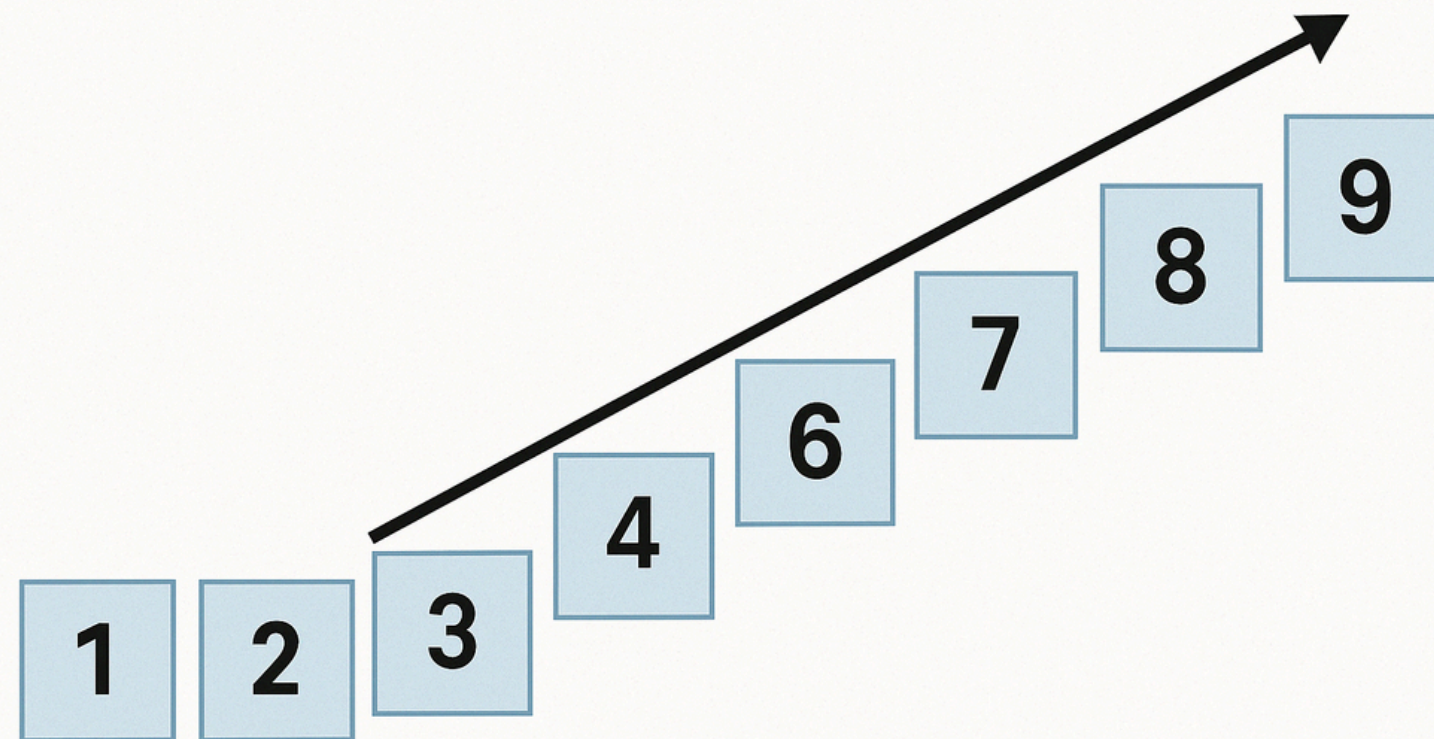
가장 긴 증가하는 부분 수열

CONTENTS

- 01 — LIS란?
- 02 — 완탐으로 풀기
- 03 — 백트래킹으로 풀기
- 04 — DP로 풀기
- 05 — 이분 탐색으로 풀기

가장 긴 증가하는 부분 수열이란

Longest Increasing Subsequence



가장 긴 증가하는 부분 수열(LIS)

배열 중 특정 인덱스를 꺼내 증가하는 수열을 만들 때 가장 길이가 긴 것을 고르면 되는 문제

- ✓ 다양한 접근법
- ✓ 알고리즘을 적용하는 사고 방식
- ✓ 기존보다 깊은 접근

01

LIS란

2	1	3	6	4	5
	1	3		4	5

02

완탐적 사고

로직

모든 부분 수열을 만들고 증가하는 수열을 탐색
그 후 가장 긴 수열의 크기를 탐색

알고리즘

DFS를 이용하여 모든 수열을 탐색

시간 복잡도

모든 값을 넣고 빼기 때문에
 $O(2^N)$

공간 복잡도

수열을 그냥 조회하면서 임시 배열에 저장
 $O(N)$

03

백트래킹적 사고

필요없는 탐색을 제거

증가하는 값으로만 탐색을 진행합니다

03

백트래킹적 사고

로직

증가하는 방향으로만 탐색합니다.

알고리즘

DFS를 이용하여 모든 수열을 탐색
But 현재보다 큰 수로만 탐색

시간 복잡도

최악의 경우인 정렬된 배열이라면
모든 값을 넣고 빼기 때문에
 $O(2^N)$

공간 복잡도

수열을 그냥 조회하면서 임시 배열에 저장
 $O(N)$

04

DP적 사고

중복 및 필요없는 탐색을 제거

**탐색된 인덱스라면
메모라이징한 값을 사용합니다.**

04

DP적 사고

로직

현재의 인덱스는 자신보다 작은 값을 가지는
인덱스의 최대값에서 1을 더한 값
그리고 DP 배열을 탐색하여 최대 값을 찾습니다.

알고리즘

DP를 사용하여 각 인덱스의 최대 길이를 구함

시간 복잡도

각 인덱스별로 이전 인덱스를 조회하기 때문에
 $O(N^2)$

공간 복잡도

DP에 사용할 배열을 추가로 사용합니다.
 $O(N)$

이분탐색적 사고

기존 문제

**현재 값보다 작은 값 중 가장 길이가 긴 값을
조회하는 것이 비효율**

이분탐색적 사고

기존 문제

현재 값보다 작은 값 중 가장 길이가 긴 값을
조회하는 것이 비효율

해결책

그럼 저장 방식을 다르게 바꿔보자
직접 가장 긴 수열을 작성하되
각 인덱스는 가능한 가장 작은 값을 넣어주기

이분탐색적 사고

해결책

그럼 저장 방식을 다르게 바꿔보자
직접 가장 긴 수열을 작성하되
각 인덱스는 가능한 가장 작은 값을 넣어주기

구현 방법

이미 만들어진 배열에서 들어갈 위치를 탐색
정렬된 배열이므로 이분 탐색으로 진행

05

이분탐색적 사고 - 완탐 ver

어렵죠?
완탐부터 다시 가봅시다

05

이분탐색적 사고 - 완탐 ver

로직

가장 긴 수열이 될 리스트를 제작합니다.

알고리즘

새로운 리스트에 들어갈 위치를 완탐으로 찾기

시간 복잡도

최악의 경우
새로 만든 리스트의 크기가 N이 되기 때문에
 $O(N^2)$

공간 복잡도

DP에 사용할 배열을 추가로 사용합니다.
 $O(N)$

05

이분탐색적 사고

정렬된 리스트에 해당 값이 들어갈 위치 탐색

이분 탐색

05

이분탐색적 사고 - 완탐 ver

로직

가장 긴 수열이 될 리스트를 제작합니다.

알고리즘

새로운 리스트에 들어갈 위치를 **이분탐색**으로 찾기

시간 복잡도

최악의 경우
새로 만든 리스트의 크기가 N이 되기 때문에
 $O(N\log N)$

공간 복잡도

DP에 사용할 배열을 추가로 사용합니다.
 $O(N)$