


Binary Search

# Index	704
📅 CreatedAt	@September 28, 2022
👤 Person	 Ally Hyeseong Kim
☀️ Status	Done
☰ Tags	Binary Search Python
📅 UpdatedAt	@September 28, 2022

References


LeetCode - The World's Leading Online Programming Learning Platform
Level up your coding skills and quickly land a job. This is the best place to expand your knowledge and get prepared for your next interview.

 <https://leetcode.com/problems/binary-search>



파이썬 알고리즘 인터뷰

2021 세종도서 학술부문 선정작. 현업과 실무에 유용한 주요 알고리즘 이론을 깊숙이 이해하고, 파이썬의 핵심 기능과 문법까지 상세하게 이해할 수 있는 취업용 코딩 테스트를 위한 완벽 가이드다. 200여 개가 넘는...

 <https://www.aladin.co.kr/shop/wproduct.aspx?ItemId=245495826>



References

- [1. 반복](#)
- [2. 재귀](#)
- [3. Python bisect](#)
- [4. Python index](#)

1. 반복

```
class Solution:
    def search(self, nums: List[int], target: int) -> int:
        left = 0
        right = len(nums) - 1
        index = (left + right) // 2
        while True:
            if left > right:
                return -1
            if target == nums[index]:
                return index
            elif target > nums[index]:
                left = index + 1
                index = (left + right) // 2
            else:
                right = index - 1
                index = (left + right) // 2
```

2. 재귀

```

class Solution:
    def permute(self, nums: List[int]) -> List[List[int]]:
        if left <= right:
            mid = (left + right) // 2

            if nums[mid] < target:
                return binary_search(mid + 1, right)
            elif nums[mid] > target:
                return binary_search(mid, right - 1)
            else:
                return mid

        else:
            return -1

        return binary_search(0, len(nums) - 1)

```

3. Python bisect

```

class Solution:
    def permute(self, nums: List[int]) -> List[List[int]]:
        index = bisect.bisect_left(nums, target)

        if index < len(nums) and nums[index] == target:
            return index

        else:
            return -1

```

- `bisect.bisect_left(list, target)` : 정렬된 `list` 에 `target` 을 삽입할 위치를 반환한다. 이미 `target` 값이 `list` 에 있을 경우, 첫번째로 `target` 과 같은 element의 위치를 반환한다.
- `bisect.bisect_right(list, target)` : 정렬된 `list` 에 `target` 을 삽입할 위치를 반환한다. 이미 `target` 값이 `list` 에 있을 경우, 마지막으로 `target` 과 같은 element의 오른쪽 위치를 반환한다.

4. Python index

```

class Solution:
    def permute(self, nums: List[int]) -> List[List[int]]:
        try:
            return nums.index(target)
        except ValueError:
            return -1

```