Problem definition:

給定一排序後的陣列 nums，欲查找一元素 target，並回傳 index，若無，回傳 -1。

Algorithm:

(1). 比較陣列中間元素，並可將原陣列切成左半右半子陣列。

(2). 若中間元素為 target，即找到 target.

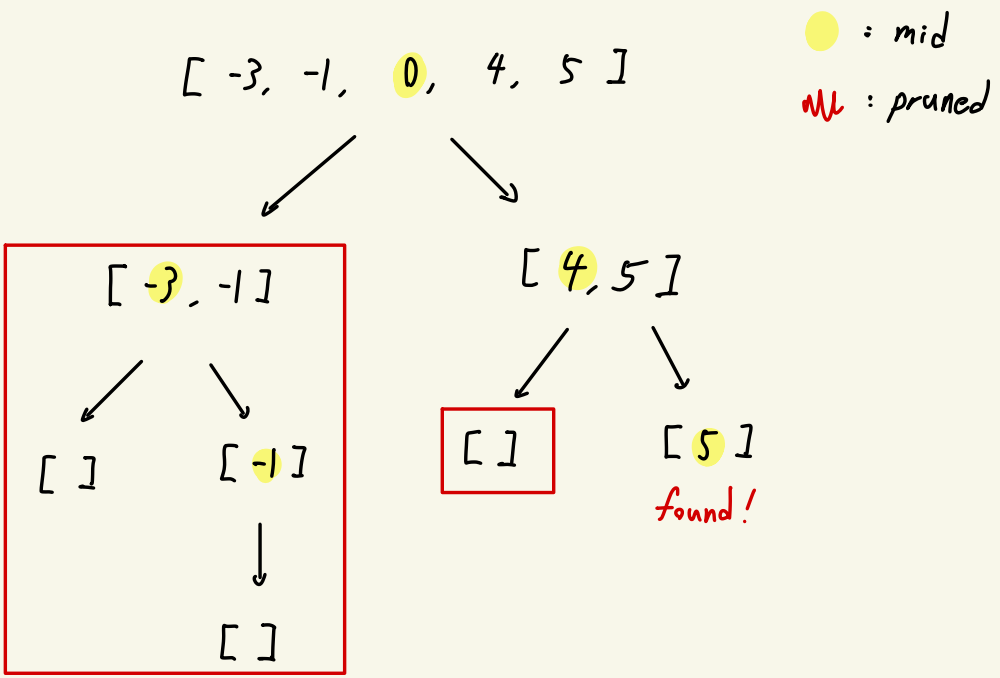
若 target > 中間元素，target 必在右半子陣列

若 target < 中間元素，target 必在左半子陣列

(3). 持續在子陣列尋找，直至找到或子陣列最終為空停止。

Example: nums = [-3, -1, 0, 4, 5]

target = 5



∃ 利用 sorted 性質，我們可以做 pruning，不用考慮其中一半子陣列 #.

Time Complexity:

$$\begin{cases} T(n) = T(\frac{n}{2}) + O(1) \\ T(1) = 1 \end{cases} \#.$$

$$\exists\ T(n) = O(\lg n)$$

Implementation Details: ① 如何 iterative 縮限 subproblem?

利用 two pointers: left & right 來控制考慮範圍.

定義方式: 1. [left, right]

2. (left, right)

是否要包含左右邊界在定義中.

eg. [-3, -1, 0, 4, 5]

0. [left, right]: [0, n-1]

0. (left, right): (-1, n)

① mid 如何定義? 如何做 rounding?

1. $\lceil \frac{left + right}{2} \rceil$

2. $\lfloor \frac{left + right}{2} \rfloor$

② 左半, 右半子陣列如何定義?

子陣列即考慮子問題範圍, 需包括要搜尋的元素們。

④. 實作時，以2個元素的陣列為 edge case 考慮。

適用範圍：

(1). 陣列：若以其它 data structure 儲存可以以別的方式 search.

2. 排序後：若未做排序，需先有排序的 overhead.