**Problem**

Given a sorted array A[1...n] of n distinct integers, you want to find out the index i for which A[i] = i if it exists. Please design a Divide-and-Conquer algorithm that runs in time O(lgn). (Analyze your algorithm and show it is correct.)

**Idea:**

Step1:檢查陣列中 中間的值是否等於他的index

Step2:若值小於index則將index以上的元素丟入divide

若值大於index則將index以下的元素丟入divide

若值相等index則將index以下跟以上的元素分別丟入divide

Step3:直到divide的array為空就返回

由於每次都取一半的array所以深度為logn，每層只比較一次所以為O(lgn)

**Pseudocode:**

Call divide(1, input\_array)

Function divide(start, arr):

if arr is not empty:

return

mid = len(arr)/2

if arr[mid] == start+mid:

print(arr[mid])

divide(start, arr[: mid])

divide(mid+start+1, arr[mid+1:])

else if arr[mid] < start+mid:

divide(mid+start+1, arr[mid+1:])

else if arr[mid] > start+mid:

divide(start, arr[: mid])