Name: Zanning Wang
zID: z5224151

4.

```
left = 0
right = pow(2,n)-2
while left < right:
    middle = Math.ceil(pow(2,n)-1)
    if middle - index_middle = 2:
        right = middle
    else:
        left = middle
return left+1
```

Because this is a consecutive positive integers list, we can use binary search to find the missing number. We can maintain two pointer called left and right, and we also have a pointer called middle, if the middle 2 smaller than current index, it means the missing number is smaller than the middle. If the middle 1 smaller than the current index, the missing number is larger than the middle. After the whole loops, we have the missing number between left and right. During this process, because the length of array is 2_n-1, therefore the time complexity is $O(\log_2 2^n) = O(n)$