

Minimum Swaps

<https://www.hackerrank.com/challenges/minimum-swaps-2/problem?>

[h_l=interview&playlist_slugs%5B%5D=interview-preparation-kit&playlist_slugs%5B%5D=arrays](#)

My approach

loop thru each number

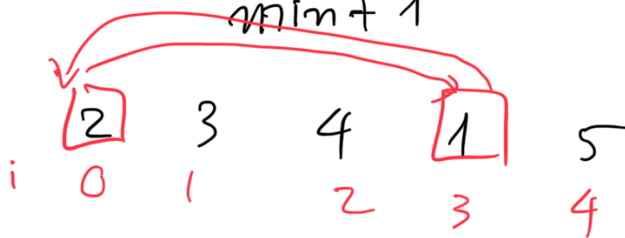
check if $ind == number \Rightarrow$ next number

else

find index of correct number

swap 2 numbers

~~min~~ + 1



find index of correct number
for each current index

↓

swap

↓

min swap + 1

Solution

```

def minimumSwapsSolution(arr):
    swap=0
    i=0
    while i<len(arr):
        if arr[i]==(i+1):
            i+=1
            continue
        arr[arr[i]-1], arr[i] = arr[i], arr[arr[i]-1]
        swap+=1
    return swap

```

similar to my approach, but:

keep swapping until the number in correct order and adding swap-times → move on next iteration

What I do wrong: move on next iteration after each swap (takes too long)