Challenge 10: Rearrange Sorted List in Max/Min Form

Arrange elements in such a way that the maximum element appears at first position, then minimum at second, then second maximum at third and second minimum at fourth and so on.

We'll cover the following

- Problem Statement
  - Input:
  - Output:
  - Sample Input
  - Sample Output
- Coding Exercise

Problem Statement#

Implement a function called `max_min(lst)` which will re-arrange the elements of a sorted list such that the 0th index will have the largest number, the 1st index will have the smallest, and the 2nd index will have second-largest, and so on. In other words, all the even-numbered indices will have the largest numbers in the list in descending order and the odd-numbered indices will have the smallest numbers in ascending order.

Input:#

A sorted list

Output:#

A list with elements stored in max/min form

Sample Input#

```
lst = [1,2,3,4,5]
```

Sample Output#

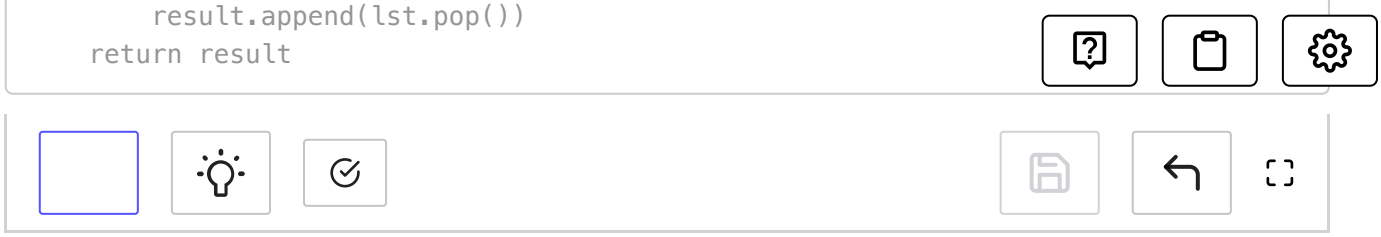
```
lst = [5,1,4,2,3]
```

Coding Exercise

Take a close look and design a step-by-step algorithm first before jumping on to the implementation. This problem is designed for your practice, so try to solve it on your own first. If you get stuck, you can always refer to the solution provided in the solution section. Good Luck!

```
def max_min(lst):
    # Write your code here
    result = []
    len_lst = len(lst)
    for i in range(len_lst//2):
        result.append(lst.pop())
        result.append(lst[i])
    if len_lst%2 != 0:
```

```
        result.append(lst.pop())
    return result
```



Interviewing soon? We've partnered with Hired so that companies apply to you instead of you applying to them. See how

Back

Solution Review: Rearrange Positive &...

Next

Solution Review: Rearrange Sorted Lis...

Completed

Report an Issue