Practice 4 Repetition + List

Note: Use code given in box for next tasks

1. Check & count, how many adjacent pairs are out of order? (out of order means, left element is larger than right element)

2. Find average of the list. Check and print, how many elements are smaller than average

```
import random as r

def init_list_random(x):
    for i in range(20):
        x.append(r.randint(10, 99))

def main():
    x=[]
    init_list_random(x)
    print(x)

main()
```

3. Find minimum element from list. Subtract it from all elements and print modified list

```
[37, 90, 42, 84, 79, 97, 34, 31, 10, 59, 83, 64, 21, 74, 15, 56, 19, 46, 25, 13]
[27, 80, 32, 74, 69, 87, 24, 21, 0, 49, 73, 54, 11, 64, 5, 46, 9, 36, 15, 3]
```

4. Select two random elements from list, and swap them. Repeat this 50 times and print list again

```
[39, 98, 96, 27, 80, 87, 43, 68, 43, 81, 63, 92, 85, 80, 91, 76, 66, 43, 97, 91]
[43, 68, 92, 80, 80, 85, 27, 76, 91, 91, 96, 39, 81, 43, 87, 43, 98, 63, 97, 66]
```

5. Check & count, how many adjacent pairs are out of order. Do apply shuffle process and count unordered pairs again and print them

6. You have unordered array, generated by the provided code. It is possible that you may have ordered sub-array within the given array. Print the longest ordered array. See example:

```
[39, 98, 96, 27, 80, 87, 43, 68, 43, 81, 63, 92, 85, 80, 91, 76, 66, 43, 97, 91]
27    80    87
```

7. Consider adjacent elements as pair and find the pair with highest sum

8. Without sorting the list print elements from smallest to largest (Again note that, you don't have to sort element, however you might use another list for your support, you may copy elements in another list but you can't sort the other list as well