

高矮个子排队

题目描述：

现在有一队小朋友，他们高矮不同，我们以正整数数组表示这一队小朋友的身高，如数组 {5,3,1,2,3}。

我们现在希望小朋友排队，以 “高” “矮” “高” “矮” 顺序排列，每一个 “高” 位置的小朋友要比相邻的位置高或者相等；每一个 “矮” 位置的小朋友要比相邻的位置矮或者相等；

要求小朋友们移动的距离和最小，第一个从 “高” 位开始排，输出最小移动距离即可。

例如，在示范小队 {5,3,1,2,3} 中， {5, 1, 3, 2, 3} 是排序结果。 {5, 2, 3, 1, 3} 虽然也满足 “高” “矮” “高” “矮” 顺序排列，但小朋友们的移动距离大，所以不是最优结果。

移动距离的定义如下所示：

第二位小朋友移到第三位小朋友后面，移动距离为 1，若移动到第四位小朋友后面，移动距离为 2；

输入描述：

排序前的小朋友，以英文空格的正整数：

4 3 5 7 8

注：小朋友<100 个

输出描述：

排序后的小朋友，以英文空格分割的正整数：

4 3 7 5 8

补充说明：

4（高）3（矮）7（高）5（矮）8（高）， 输出结果为最小移动距离，只有 5 和 7 交换了位置，移动距离都是 1。

示例 1

输入：

4 1 3 5 2

输出：

4 1 5 2 3

说明：

示例 2

输入：

1 1 1 1 1

输出：

1 1 1 1 1

说明：

相邻位置可以相等

示例 3

输入：

xxx

输出：

[]

说明：

出现非法参数情况， 返回空数组

```
import java.util.Scanner;
```

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        Scanner scanner = new Scanner(System.in);
```

```
        try{
```

```
            String inputStr = scanner.nextLine();
```

```
            String[] s = inputStr.split(" ");
```

```
            int[] a = new int[s.length];
```

```
            for(int i = 0 ; i < s.length; i++){
```

```
                a[i] = Integer.parseInt(s[i]);
```

```
            }
```

```
            int i = 0;
```

```
            while(i + 1 < a.length){
```

```
                if((i+1) % 2 == 0){
```

```
                    if(a[i] > a[i+1]){
```

```
                        int x = a[i+1];
```

```
                        a[i+1] = a[i];
```

```
                        a[i] = x;
```

```
                    }
```

```
                }else{
```

```
                    if(a[i] < a[i+1]){
```

```
                        int x = a[i+1];
```

```
                        a[i+1] = a[i];
```

```
        a[i] = x;
    }
}
i++;
}
StringBuilder sb = new StringBuilder();
for(int j = 0 ; j < a.length ; j++){
    sb.append(a[j] + " ");
}
System.out.println(sb.toString());
    }catch(Exception e){
System.out.println("");
}
}
}
```