

最小循环子数组

题目描述：

给定一个由若干整数组成的数组 `nums`，请检查数组是否是由某个子数组重复循环拼接而成，请输出这个最小的子数组。

输入描述：

第一行输入数组中元素个数 `n`， $1 \leq n \leq 100000$

第二行输入数组的数字序列 `nums`，以空格分割， $0 \leq \text{nums}[i] < 10$;

输出描述：

输出最小的子数组的数字序列，以空格分割；

补充说明：

数组本身是其最大的子数组，循环 1 次可生成的自身；

示例 1

输入：

9

1 2 1 1 2 1 1 2 1

输出：

1 2 1

说明：

数组[1,2,1,1,2,1,1,2,1]可由子数组[1,2,1]重复循环 3 次拼接而成

```
n = int(input())
```

```
arr = input().split(" ")
```

```
if arr.count(arr[0]) == n:
```

```
    print(arr[0])
```

```
    con = 0
```

```
else:
```

```
    con = 1
```

```
k = n-1
```

```
while con:
```

```
    long = n/k
```

```
    if int(str(long).split(".")[1]) != 0:
```

```
        k -= 1
```

```
        continue
```

```
    x = int(str(long).split(".")[0])
```

```
    a = arr[0:x]
```

```
    b = arr[x:x*2]
```

```
    c = arr[n-x:n]
```

```
    if a == b and a == c:
```

```
        print(" ".join(a))
```

```
        break
k -= 1
if k == 0:
    print(" ".join(arr))
    break
```