

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， $0 \leq nums[i] < 10$;

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

```
import sys
```

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

```
nums = list(map(int,input().split()))
```

```
for i in range(1,n+1):
```

```
    if n%i!=0:
```

```
        continue
```

```
    else:
```

```
        t=int(n/i)
```

```
        if nums == nums[0:i]*t:
```

```
            for j in nums[0:i]:
```

```
                print(j,end=' ')
```

```
            break
```