

原始代码：

```
1 print(*map(lambda x: ' '.join(map(lambda
  y:str(y),x)),sorted(set(__import__("itertools").permutations([int(i) for i
  in [input(),input().split(' ')[1]])))),sep='\n')
```

翻译成成人话：

```
1 import itertools
2 num=[int(i) for i in [input(),input().split(' ')[1]]
3 ans=sorted(set(itertools.permutations(num))) ## 全排列，去重，排序
4 # [(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1)]
5 print(*map(lambda x: ' '.join(map(lambda y:str(y),x)),ans),sep='\n')
6 # 里层map,将(1,2,3)的每一个元素变成字符串
7 # [(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1)] -->
  [('1','2','3'),....]
8 # 外层map,将('1','2','3')连接成'1 2 3'
9 # print(*iteraoctr),迭代器里面一个个输出元素,以sep分割
```

模块

import() 函数用于动态加载类和函数。

如果一个模块经常变化就可以使用 **import()** 来动态载入。

python的模块导入十分简单

同一目录下有

my_module.py

test.py

2个文件

test.py内容：

```
1 import my_module
2 print('当前name:',__name__)
```

my_module.py内容

```
1 print('hcx真帅')
2 print(__name__)
```

test.py执行结果

```
1 hcx真帅
2 my_modue1
3 当前name: __main__
```

运行即执行my_module.py所有代码一遍

itertools.permutations

python itertools库的Permutations函数

对字符串或者数字，生成全排列

itertools也有生成组合的函数

map

map() 会根据提供的函数对指定序列做映射。

```
1 | map(function, iterable, ...)
```

函数式编程：不用变量保存状态，不可修改变量

map,filter,reduce

function的效果会作用到每个元素上

```
1 | map(lambda x, y: x + y, [1, 3, 5, 7, 9], [2, 4, 6, 8, 10])
```

[3, 7, 11, 15, 19]

lambda

匿名表达式，不需要指明函数名的函数，即开即用

```
1 | lambda x:x+1
2 | # 等价于
3 | def f(x)
4 |     x+1
```