

## 4.15 顺序迭代合并后的排序迭代对象¶

### 问题¶

你有一系列排序序列，想将它们合并后得到一个排序序列并在上面迭代遍历。

### 解决方案¶

`heapq.merge()` 函数可以帮你解决这个问题。比如：

```
>>> import heapq
>>> a = [1, 4, 7, 10]
>>> b = [2, 5, 6, 11]
>>> for c in heapq.merge(a, b):
...     print(c)
...
1
2
4
5
6
7
10
11
```

### 讨论¶

`heapq.merge` 可迭代特性意味着它不会立马读取所有序列。这就意味着你可以在非常长的序列中使用它，而不会有太大的开销。比如，下面是一个例子来演示如何合并两个排序文件：

```
with open('sorted_file_1', 'rt') as file1, \
    open('sorted_file_2', 'rt') as file2, \
    open('merged_file', 'wt') as outf:

    for line in heapq.merge(file1, file2):
        outf.write(line)
```

有一点要强调的是 `heapq.merge()` 需要所有输入序列必须是排过序的。特别的，它并不会预先读取所有数据到堆栈中或者预先排序，也不会对输入做任何的排序检测。它仅仅是检查所有序列的开始部分并返回最小的那个，这个过程一直会持续直到所有输入序列中的元素都被遍历完。