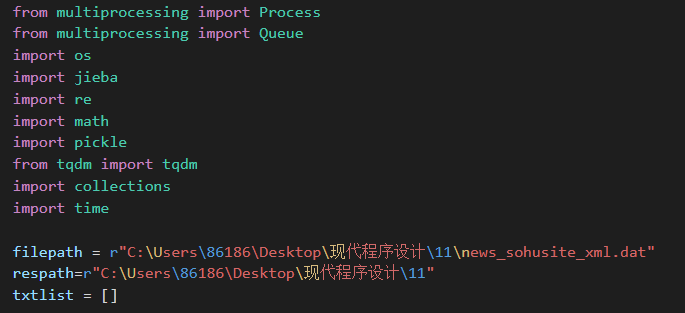
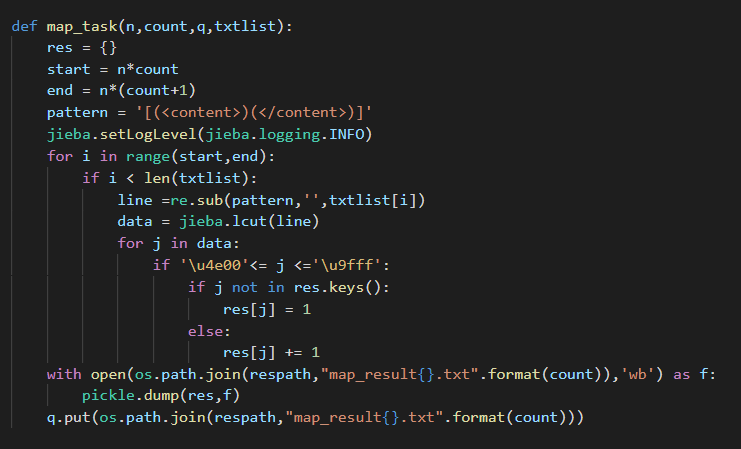
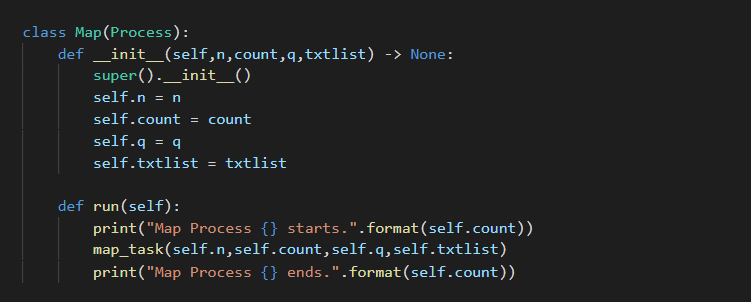
MapReduce是利用多进程并行处理文件数据的典型场景。作为一种编程模型，其甚至被称为Google的”三驾马车“之一(尽管目前由于内存计算等的普及已经被逐渐淘汰)。在编程模型中，Map进行任务处理，Reduce进行结果归约。本周作业要求利用Python多进程实现MapReduce模型下的文档库（***搜狐新闻数据(SogouCS)，下载地址：https://www.sogou.com/labs/resource/cs.php***），注意仅使用页面内容，即新闻正文）词频统计功能。具体地：



1. Map进程读取文档并进行词频统计，返回该文本的词频统计结果。

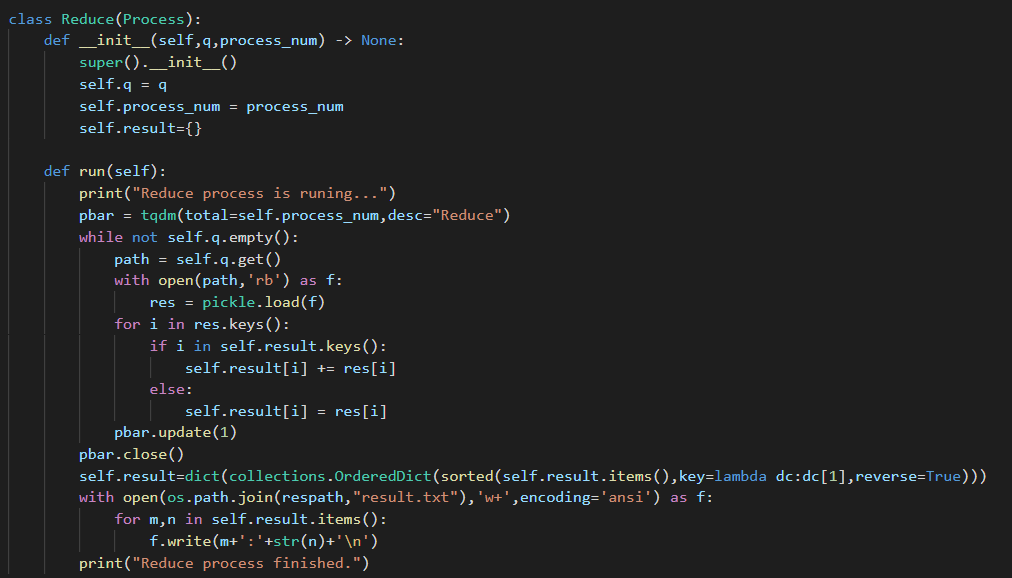
**利用正则表达式对于新闻信息进行处理，只保留正文部分；然后进行分词统计词频，将结果保存到文件中，在队列中放入文件路径**





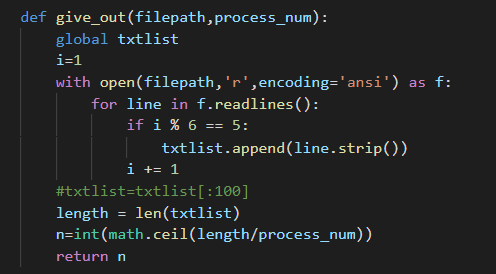
1. Reduce进程收集所有Map进程提供的文档词频统计，更新总的文档库词频，并在所有map完成后保存总的词频到文件。

**利用队列中的路径更新总的字典，并将结果输出到文件中**



1. 主进程可提前读入所有的文档的路径列表，供多个Map进程竞争获取文档路径；或由主进程根据Map进程的数目进行分发；或者单独实现一个分发进程，与多个MAP进程通信。

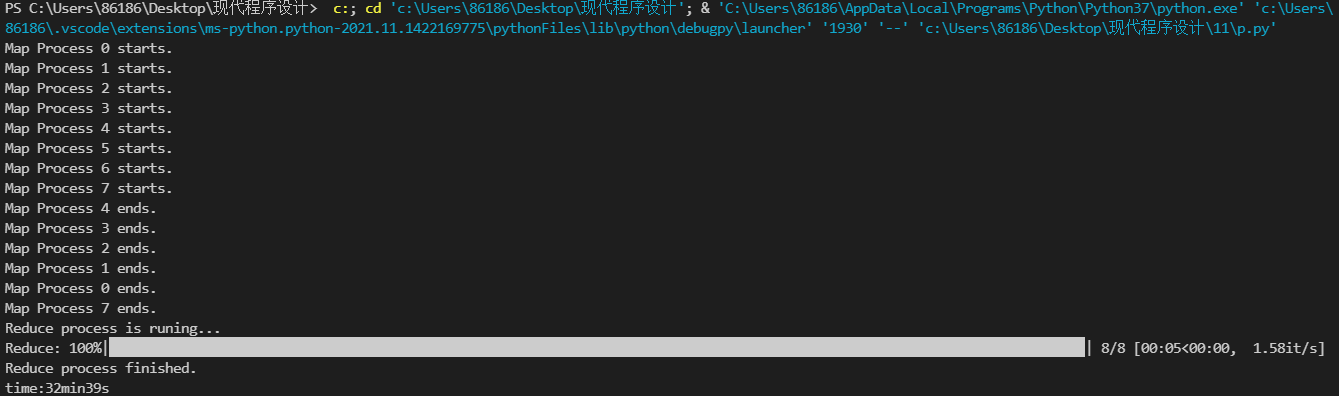
**主进程提前对信息进行处理，只保留<content>行；并根据进程数目得到每个进程要处理的新闻数量**

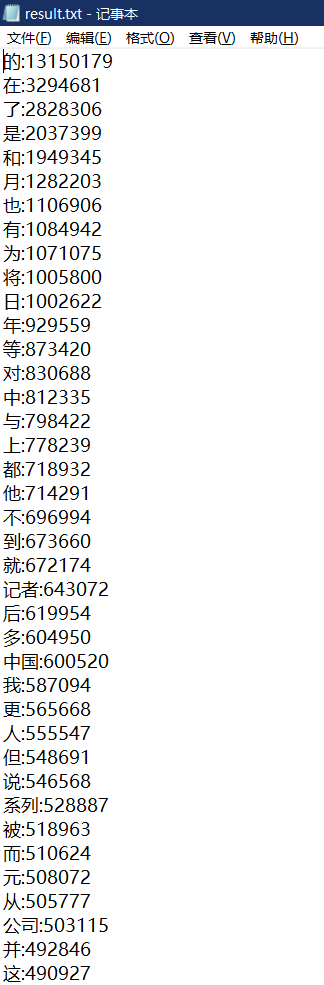


Main函数：



运行结果：





1. 记录程序运行时间，比较不同Map进程数量对运行时间的影响，可以做出运行时间-进程数目的曲线并进行简要分析。

随着进程数目增加运行时间显著减少，担当进程数目多于8时，时间并未减少反而有所增加。这是因为电脑的cpu数量为8，最多只能同时运行8个进程，而设定的进程数目多于8个时，部分进程并不能同时运行，而是在等待前面的完成才能运行。



