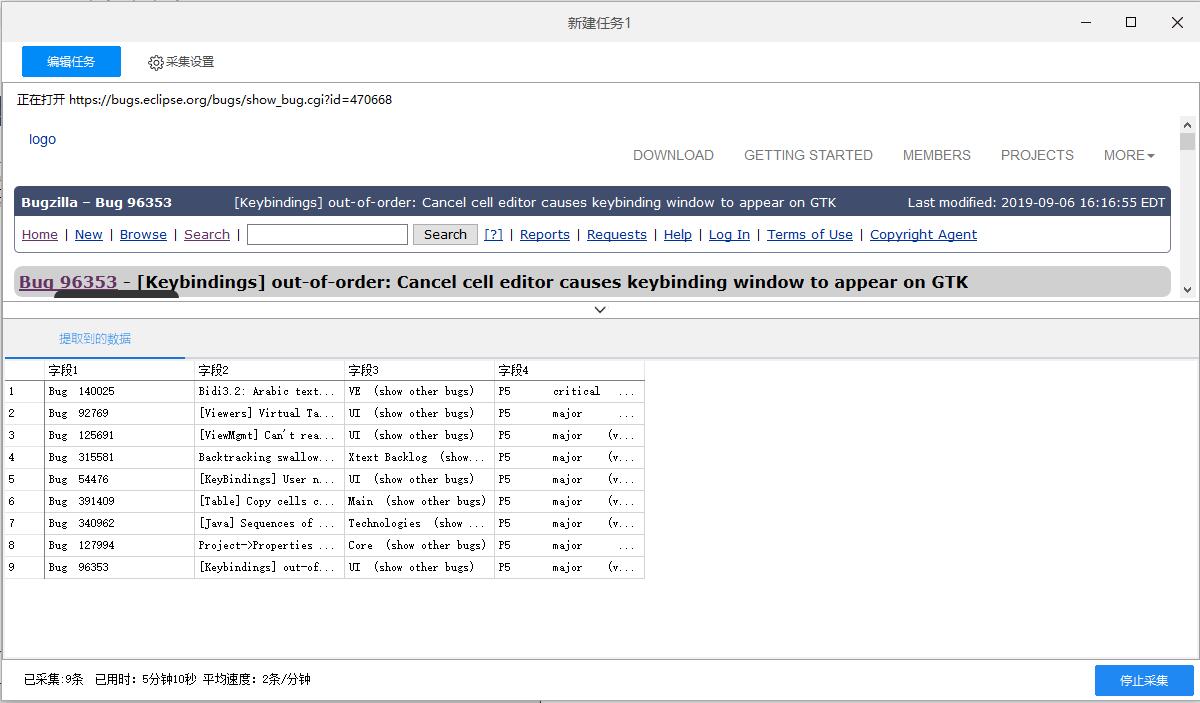
**实验二：软件需求的优先级排序**

1. 实验目标

为软件需求进行优先级排序

1. 实验过程

**1.数据获取**

利用爬虫软件，在Ecilipse的缺陷报告网站（https://bugs.eclipse.org/bugs/）爬取缺陷报告记录，将它们视为软件需求。共爬取了5000多条记录，爬取的字段有：标识号ID、组件Component、总结描述Summary、缺陷等级Importance。

**2.需求排序：**

①结合Bugzilla的需求优先级分级标准和实际情况，制定五个优先级Highest、High、Medium、Low、Lowest，并列出属于每个优先级的关键词，如下：

* Highest
* 内存泄露：memory leak、out of memory
* 数据丢失、损坏：data loss、destroy
* 无法运行、不工作：not work、cannot run
* 死循环
* ….
* High
* 文件相关错误：file
* 刷新相关错误：refrsh
* 同步相关错误：[Synchronous](javascript:;)
* ….
* Medium
* 计算错误：calculation error
* 编译错误：compilation error
* 功能错误： function errors
* 段错误：seg fault
* ….
* Low
* 性能 performence
* Lowest
* 建议、优化： consider、want、Optimized
* 界面问题：
  + - 菜单 menu
    - 对话框：diogram
    - 拼写：spelling
    - 字体： font

② 利用excel自带的函数IFERROR、SERACH、LOOKUP对需求按关键词进行排序：对每一条记录的Summary字段，按优先级从低到高的顺序查找关键词，若包含某一优先级的关键词，则将需求归为该优先级，继续对下一条记录分级；否则查找下一优先级的关键词。

注：

* + 1. 考虑到同一关键词的不同语法形式应属于同一优先级，我们使用了Search函数。Search函数是模糊查找，不区分大小写，且可使用通配符\*和?。
    2. 若同一条记录里含有多个优先级的关键词，则由于查找顺序是由低到高，它最终将被归为查找到的最高优先级。

**3.方法验证**

①首先把爬取的缺陷报告里的Priority和我们自己设的Priority从String转化为数值，并分别存储在Official\_Priority和Set\_Priority两列中。

用到的函数有：

1. =IFERROR(LOOKUP(9^9,SEARCH({"Highest","High","Medium","Low","Lowest"},F3),{"1","2","3","4","5"}),"")
2. =IFERROR(LOOKUP(9^9,SEARCH({"P1","P2","P3","P4","P5"},C3),{"1","2","3","4","5"}),"")
3. =ABS(G3-H3)

②然后，对每条记录取这两个数值的差的绝对值，作为它的Accuracy，用来衡量我们设的优先级与标准的Ecilipse定的缺陷等级的差距，从而评判我们的优先级分级方法是否合理。

③最后，计算Accuracy的平均值0.585063526、方差0.511283342、最大值4。可见，我们的分级方法较为合理。