文本查找工具

摘要：在文本文件中查询子串是否存在，子串及文本文件的内容均从键盘输入，Windows 从键盘输入 CTRL+Z（值为 −1）表示文件结束，只考虑英文字符。

1. 任务

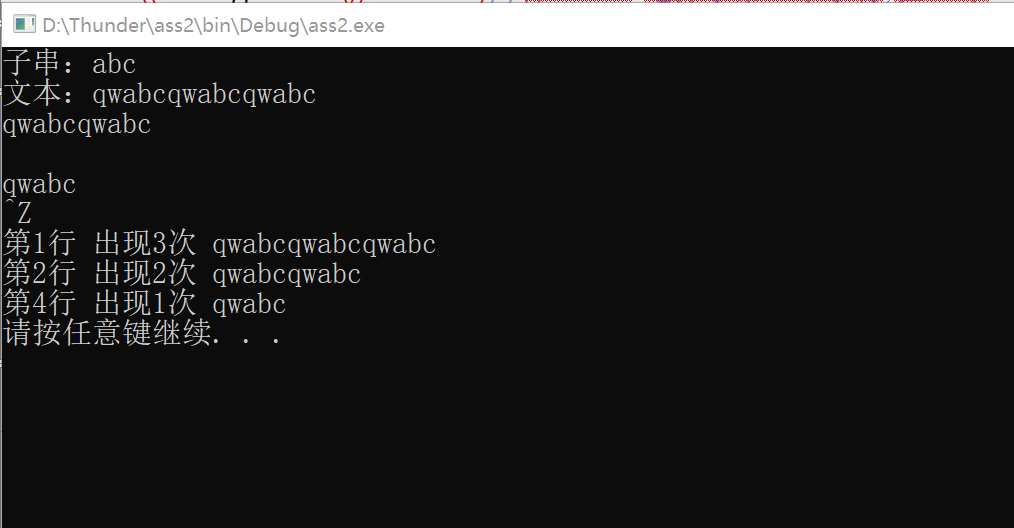
**文本查找工具**

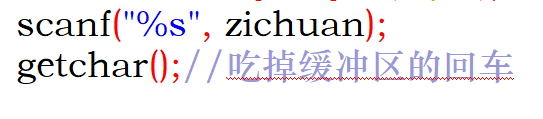
在文本文件中查询子串是否存在，如果存在，输出行号、该行字符串及子串在该行中出现的次数。待查询的子串及文本文件的内容均从键盘输入。输入的第一行为子串，后续多行为文本文件的内容，Windows、DOS 中从键盘输入 CTRL+Z（值为 −1）表示文件结束\*\*，Linux/Unix中用 CTRL+D（其值同样为 −1）结束。为了避免不同的字符编码带来的困扰，此处只需考虑英文字符即可。

1. 基本思路

目标子串和文本文件都需要存放在字符数组里；每行的字符可以定义一个结构体，记录每行中字串出现的次数；可以判断是否输入换行符来记录行数；但因为getchar会吃掉每行第一个字符，所以要定义一个拼接容器来拼接第一个字符和后续字符串。（后来加入了对空行的处理，就是无需拼接，直接把\n赋给二维数组并且行数加一，用continue跳过后续语句）

1. 测试结果



1. Pains and Gains
   1. 输入子串后会输入回车，也会被记为一个字符，所以要用

把回车吃掉

* 1. 文件结束的标志CTRL+Z用EOF表示
  2. 

第一个字符会被存在ch里，从第二个字符开始进入循环，所以每行第一个字符要单独赋值到二维数组的拼接容器里 

* 1. sizeof的返回值包括’\0’，strlen的返回值不包括’\0’
  2. 将结构体struct Text text[100]中的cishu(次数)变量全部初始化为0后，text[100].cishu及往后的结构体，虽然超出定义范围，但cishu变量仍为0
  3. Strcat(str1,str2)会将str2拼接到str1上（覆盖str1的’\0’）
  4. strncmp(str1,str2,len)，其中len是字符串长度（不包含结尾’\0’）