题目要求：

本题的目标是在例子程序的基础上，仅仅修改main 函数，要求达到的目的是不论输入的文本有多长，都能正确输出长度（即便文本的长度超出了数组的上限），并且能够尽可能多地输出文本本身

解：尽可能输出全部的本的内容只有通过修改数组上限来改编，因此本题的难点是如何正确输出理论上无限长的字符串的字符长度。

实现方法：

1. 设置一个临时数组，从文本开头尽可能地保存文本内容，因此要判断是否为文本开头。
2. 在遇到\n之前（也就是没有遇到\n），用一个变量累加计算字符串的长度。在不断累加的过程中，反复将max和这个累加变量对比，如果累加变量大于max，则max等于累加变量。
3. 直到最后遇到\n，累加上本行最后输入的一串字符长度，对比max，如果累加变量大于max，让max=累加变量，并且复制这个最长行的临时数组为最长数组。然后这个累加变量归零，准备为下一行字符串服务。

疑问：红笔部分的位置，既然都还没有遇到\n，都还没有到一行的结尾，为什么要急着对比这一行和最长行的长度呢？暂时还没弄懂。做过实验，也没有发现有什么不同

**Line末尾是否遇到\n?**

**Yes** **No**

是否在字符串中 是否在字符串中

本行结束 本行结束， 在本行 本行开头

上行为空 未结束

If(max<prevmax+len) if(max<len) prevmax+=len;

Max=prevmax+len; max=len;

Cpystr(longest,line) cpystr(longest,temp) cpystr(temp,line)

If(max<prevmax)

Prevmax归零 max=prevmax

为下一行累积计算服务 标记状态为在一行中

标记状态为一行结束

最后输出max和longest[ ]