**实验报告**

专业：计算机科学与技术

姓名： 蔡雨谦

学号： 3210102466

日期： 2022.4.23

课程名称：C程序设计专题 指导老师： 翁恺 成绩：

实验名称：作业2： 行编译器

一、实验题目要求

实现一个GUI窗口内的单行的文本输入编辑器。

支持左右方向键、回退和删除键、ins切换插入和覆盖状态，编辑过程中有光标闪烁，插入和覆盖状态的光标形状不同。

回车后，结束输入，将输入的内容在标准输出输出。

二、实验思路和过程描述

1、main.c中注册了三个回调函数：mouselistener用来将光标移动到鼠标点击的位置。keylistener用于实现回退、删除、光标移动、切换插入与覆盖模式、输出功能。Charlistener用于实现输入功能。

2、a.h存放所用函数原型。

3、a.c：

创建了一个100容量的字符串text[100]，每个字符的宽度为7（7指光标宽度设置为7时恰好能够覆盖住一个字符，但textsize设置的是12，没找到数值间的关系）；

定义了若干变量：

position=0：当前操作的字符串部位，position\*7为光标位置；

len=0:字符串长度；

flag=0：按insert键的次数；

state=1：当前输入模式（1为插入，-1为覆盖）

（1）mouselistener（int x,int y,int button,int event):

当点击鼠标左键(button==1&&event==0)且x轴方向上未超过字符串时(x/7<=len)，光标移动到字符串上与鼠标点击位置的水平位置相同的位置（position==x/7；SetCaretPos(7\*position,1)）。

（2）charlistener (char c):

当敲击键盘上具有输出效果的按键时，在图形界面上显示输入后的界面。根据state的符号，输入分为两种模式。覆盖模式下，当前position处的字符变为c，position后移一位，当position位于末尾时，len还要+1。插入模式下，要先将position之后字符后移一位，当前position变为c，len要+1，position后移一位。

（3）keylistener(int key, int event):

按下左键且position>0时position--,按下右键且position<=len时position++。

按下enter键将text输出。

按下insert键后flag++，state=pow(-1,++flag)。如第一次按下，flag变为1，state变为-1,输入模式变为覆盖。

按下backspace，从position处起到末尾的字符全部左移一位，覆盖掉position的前一位，然后len--,position--。

按下delete，从position的下一位起到末尾的字符全部左移一位，覆盖掉position处的字符，然后len--,position不变。

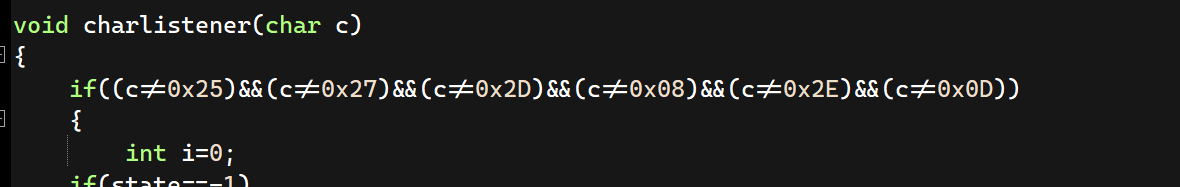
（4）设置光标

清屏；

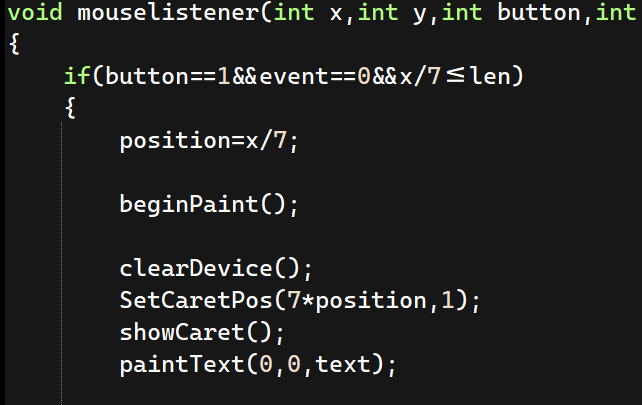
State==1时光标宽度为1，state==-1时光标宽度为7。光标位置为7\*position；

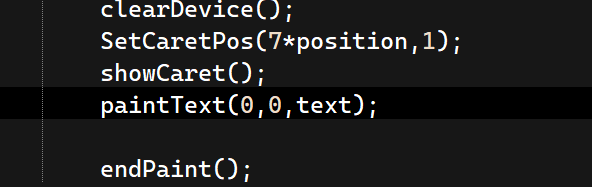
将字符串输出到图形界面。

三、实验代码解释

1、

当按下的是回车、删除等键时，会出现乱码和空格，所以在函数中加了一行判断。但是不清楚原因

2、如果光标位置直接设置在x的位置，当鼠标点击的是字符中心时，光标会出现在字符中心影响美观，且覆盖模式下不好分辨当前操作的是哪个字符。先令position=x/7取整再令光标位置为7\*position可以让光标自动处于字符间的位置。

四、实验体会与心得

1、光标的尺寸在一个函数中设置之后，到了另外一个函数中好像也能继承。如图是mouselistener中的一段代码，并没有设置光标的宽度，但是能够沿用keylistener中设置的宽度。

2、大写字母的ASCII码就是键盘字母对应的虚拟键码

3、chareventcallback函数是可以读到回车、回退、shift、ctrl等没有实际输出的键的，此时c的值为对应的虚拟键码。

4、虚拟键码为十六进制的数，将其换算成其他进制的数同样有效。