

# 浙江大学实验报告

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( $x/7 \leq len$ )，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、

```
void charlistener(char c)
{
    if((c!=0x25)&&(c!=0x27)&&(c!=0x2D)&&(c!=0x08)&&(c!=0x2E)&&(c!=0x0D))
    {
        int i=0;
        if(state== -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。

```
void mousetlistener(int x,int y,int button,int
{
    if(button==1&&event==0&&x/7<=len)
    {
        position=x/7;

        beginPaint();

        clearDevice();
        SetCaretPos(7*position,1);
        showCaret();
        paintText(0,0,text);
```

### 四、实验体会与心得

```
clearDevice();
SetCaretPos(7*position,1);
showCaret();
paintText(0,0,text);

endPaint();
```

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。