

Oslab 0实验报告

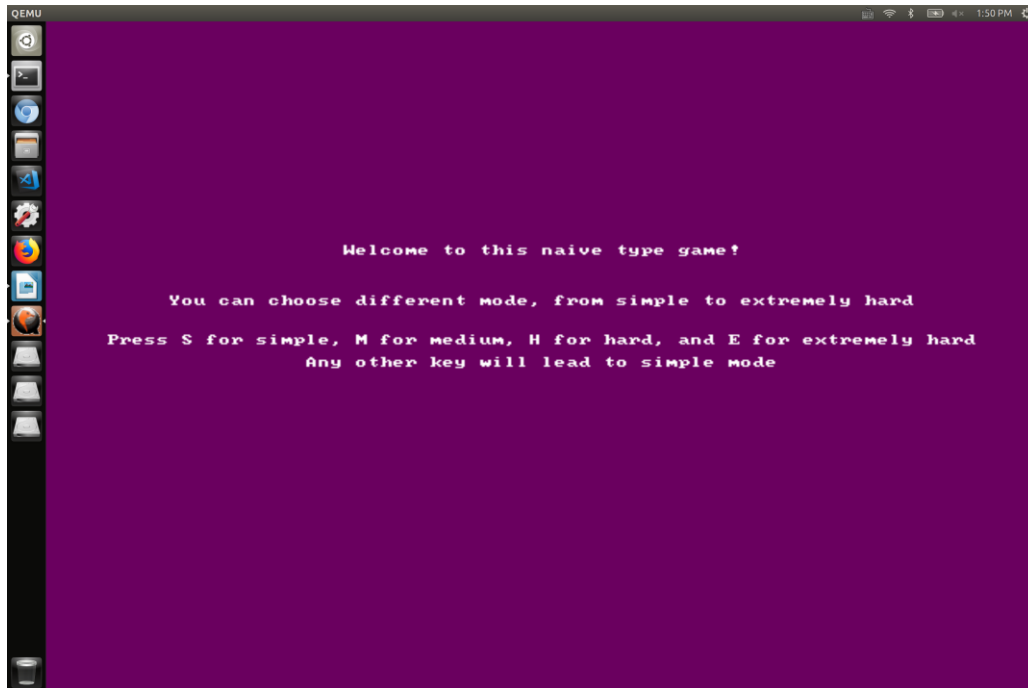
161180162 许致明

游戏内容

这是一个非常简单的打字游戏，屏幕上随机下落26个英文字母，在键盘上敲击相应的按键后，字母就会上升，从窗口上部离开屏幕。若没有及时敲击，字母就会从窗口下部离开。

游戏窗口如下图所示：

1.进入时的界面，这里可以选择难度（后面会解释）



2.游戏进行时的界面



其中，左上角显示当前的帧数，最高为30，在性能不足的情况下可能有所降低；右上角是输入成功的字母个数；右下角是输入失败的字母个数；左下角是游戏的倒计时，这个游戏会运行一分钟的时间，一分钟后，退出循环并返回到 main，最后 exit(0)。倒计时的数字会随着时间减少，由绿色逐渐变为黄色，最终变为红色。在最后10秒内，这个数字将会闪烁。

3.几种难度模式

(1) 简单(S)

此模式下，下落的均为大写字母，且速度为2像素/秒，且保持不变；

(2) 中等(M)

此模式下，下落字母速度随机取{1,2,3,4}像素/秒中的一个，其他与 S 相同；

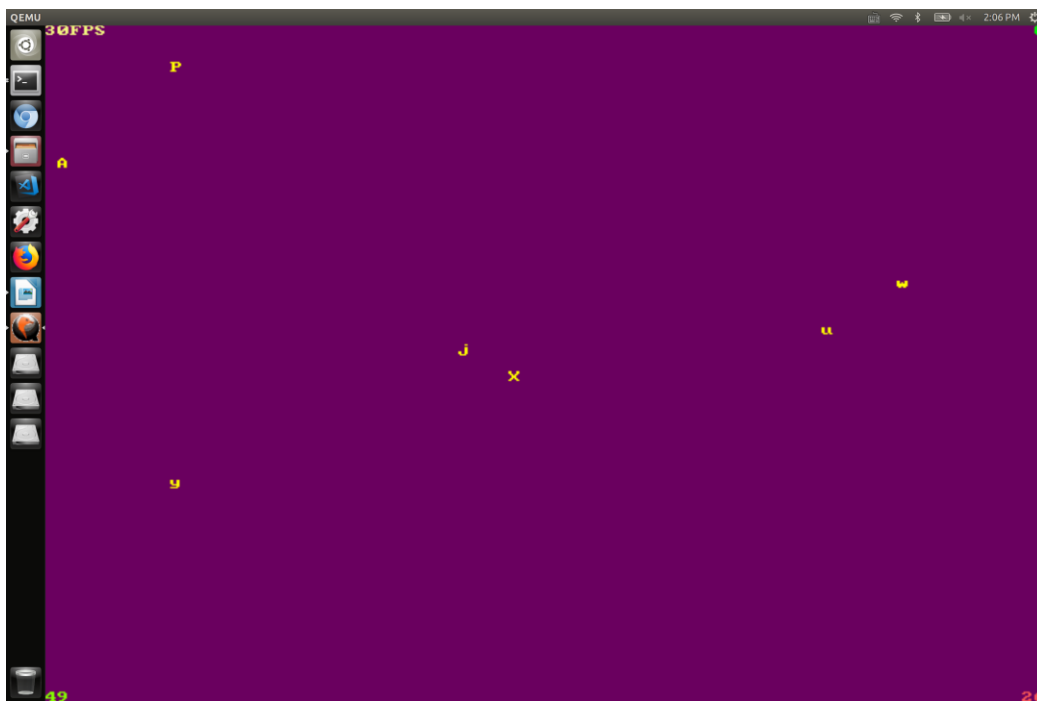
(4) 困难(H)

此模式下，字母的大小写随机生成，且在整个下落过程中保持不变，其他与 M 相同；

(3) 最难(E)

此模式下，字母的颜色会不断变化，即每帧都任取{R,Y,B,W}中的任意一种颜色，其他与 H 相同。

4.E 模式的运行截图



检查函数返回值

在写扫描设备的函数时，把循环体里的变量弄错了，结果没有扫描，而是直接使用_device 这个 API 询问了一个不存在的设备，然后我的代码里又没有检查它的返回值，直接使用了返回的空指针，导致程序不断的黑屏重启。

通过这个 bug 我体会到了 unused return value 可能带来的隐患，以后会注意那种看起来一定不会出错的函数的返回值(system(...), fgets()等)，消除这种本应很容易发现的 bug