此后如竟没有炬火 我便是唯一的光

(超详细) ncurses 库使用介绍: 实现终端 GUI 这次 ENGG1340 课程的 group project 是设计并实现一个 **text-base game**,作为跨镜上运行的 text-base game,有一个出彩的 GUI 瑜克是一个加分项

USUNIALET INDIGHT (COURSELL) 库这么一种奇的灰西、最重要的是、虽然它不离子 C++ 标准库,但是在学校 server 里然然默上有联子了《阿默杜伯密的旅程》为 都能够多种"加速物界",例如 My 、 SL 被刑罚 Foruse 花了一个下午学习了一下阻走。在这里简单的战士下,并且附上一直简单 GUI 组成的实例

Installation

在终端上输入命令 sudo apt-get install libncurses5 进行安装

Compilation

所有使用了 <ncurses.h> 座的程序,在論理时需要添加参数 -l ncurses 使编译器解接到 nc 例: g++ -o test test.cpp -lncurses

初始化窗口对象

初始化过后,我们接下来将与该窗口对象进行交互: std::cin/out, scan/printf 等标准输入输出将失效

返回常规终端模式

当使用完 ncurses,想回到常规终端模式时,使用 endwin() 来关闭窗口

输入/输出

在调用 endwin() 之前,标准输入输出将失效

想要与初始化后的窗口对象交互,我们需要使用 ncurses 库提供的输入/输出函数

由于 qp 中使用了很多 string,我自定义了两个接口用来处理 string 类型的输入输出

string instr() { char s(1024); char s(1024); char s(1024); char scann("有"(n)", s); // 这里对标的是 getline, 固年对抗策谈入 return string(s);

在指定位置输出

在 ncurses 模式中,我们可以通过 mv() 函数轻松控制光标的位置,从而实现在终端的指定位置输出 mv(x,y) 将光标移动到当前面口的第 z 行第 y 列 (万列均从 开始),且接下来输出的内容都将从该位置开始 以下实例将在从解幕中央开始 (注意,是开始而不是位于) 输出 "Hello World!"

...
int scrline, scrCol;
getmaxyx(stdscr, scrline, scrCol); // 接取标准屏幕的行/列
move (scrline / 2 - 1, scrCol / 2 - 1); // 特先标格证用基中央
print("Holl Worldf");

移动光标和論出可以在同一个函数 myprintw() 中完成: 例如上例程序可写成 myprintw(scrLine / 2 - 1, scrCol / 2 - 1, "Hello World!")

在绘出宗华后,使用 refresh() 进行副新,将绘出显示到当前屏幕上

创建新窗口/子窗口

在李指定的情况下 initscr() 初始化的商口基坛准屏幕 strisc

而有时我们需要多个窗口;通过创建新窗口 (newwin) 或子窗口 (si 创建新窗口时,其会被分配一个新的内存;而子窗口与其父窗口共用内存

WINDOM* newwin(int line, int col, int x, int y); // 创建新窗口 WINDOM* submin(WINDOM* parent, int line, int col, int x, int y); // 创建子窗口,其父窗口为 parent

删除新窗口/子窗口

翻除窗口 win 并不代表 屏幕上 win 输出的内容会消失,因此在删除之前调用 wclear(win) 与 wrefre:

不要删除 ncurses 的默认窗口 stdscr, 结束它使用 endwin() 即可

非标准窗口的交互操作

級級简单層懂,之前我们介绍的所有对 stdscr 的函数,加上前堰 w,再添加对应窗口的部针作为参数就得到了与非标准窗口交互的函 ***

---例如:printw() 对应 wprintw(win), refresh() 对应 wrefresh(win), mvprinrw() 对应 mvwprintw(win) (这个位置稍有不同)

NINDCH* win = neewin(30, 30, 5, 5);
wespeciatorium, 1, 1, "Sober*); // 注意·范围的 (1, 1) 是对节围口 win 的相对信息,而不是对于 stdeer 的 处对任意
wesfresh(vin); // 沒有學歷.使物學很多訓練展上 or
WINDOW* sub = subwin(atdser, 30, 30, 0, 0); // sub 是标准窗口的子窗口
wwwprintw(sub, 1, 1, "Skull"); wewprintw(sub, 1, 1, "Skull"); touchwin (stdscr); // 由于 sub 与标准窗口具用内存, 我们用 touchwin() 函数标 记标准窗口被移改即词,而非使用 wrefresh()

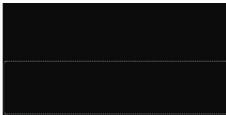
子窗口与父窗口

当我们使用 wmove() 或其他对非标准部口的交互函数对子衙口进行操作时,父爵口 (或其他子衙口) 输出的内容合管时消失 解决方法是:在 touchwin(fascr) 过后,再对父帝口进行翻新

即 wrefresh(father_scr) 或 refresh() (当父體口是标准窗口时)

WINDOW* upwin = subwin(stdscr, scrline / 2, scrCol, 0, 0); WINDOW* downwin = subwin(stdscr, scrline / 2, scrCol, scrline / 2, 0);

Satill refresh() 除t upwin 每日的输出内容 (图 box 绘出的边框) 格不会局部



昵称: 四季夏日天下第一 回龄: 5年4个月

找找看

\$8000EEC 我的参与

我的标签 思维(12) 位运算(9) 歌学(8) 最近公共相先(6

随笔分类 ENGG1340 Computer programming II(1)

随笔档案 2023年5月(2) 2023年4月(6) 2023年2月(5) 2022年12月(1 2022年10月(1) 2022年9月(3) 2022年8月(1) 2022年6月(5) 2022年5月(6) 2022年4月(6)

2. 微积分 | 笔记(637)

2. HNOI 2017 怼大佬(2)

4. 51nod 1835 完全图(1)

推荐排**行**榜 1. (超详细) ncurses 库使用介绍: 实现终 GUI (4)

1. Re:51nod 1835 完全图

庄明太强了 爱你的 3. Re:11.9 AM 洗衣机 4. Re:11.9 AM 树上的奖牌

```
窗口转储
这在实现 text-base game 中是一个很实用的功能,它可以用来实现场景的切换,与读/存档功能
例如,"开始"界面可以进入"游戏"界面,但从"游戏"界面返回时,由于屏幕已经被"游戏"界面所在窗口覆盖,我们需要重新加载"开始"界面
此时,利用鹽口的转锗,我们可以直接恢复之前的界面
标准窗口转储
标准窗口 stdscr 可以利用下面的两个函数进行存储与读取:
  int scr_dump(const char*); // 参数是当前目录下文件的名称; scr_dump 符金尼标准窗口中的内容存储到对标名称
的文件中
 的文件中

// 看当前目录下没有该各种文件、符金的键一个
int scr_restore(const char*); // scr_restore 面数符合该数可比文件中存储的内容,并将其收入到标准窗口中
以下是一个简单的例程 (gp 中的界面转换可以以此为基础)
 printw("Main Menu\n"); // 标准窗口作为主界面 main menu
refreah();
scr_dump("a"); // 链存当款标准窗口的内容
 getch();
WINDOM* win = neswin(scrline, scrcol, 0, 0);
wprintw("Came Started(\n"); //一个与标准窗口相同大小的新窗口 win 作为游戏界面
wenfreak(win)

    OpenCh()

    welcase(rolar)

    welcase(rolar)

    // 遊戏序面比其、何算高清室

    debain(rolar)
    // 粉波索母非而信息口 via 的内存

    // OpenCh()
    // 美国主角器 main.memp. 使用 sor_restore() 概定

    setramb()
    // 美国主角器 main.memp. 使用 sor_restore() 概定

若有时我们需要智时退出 ncurses 模式,回到行缓冲模式,我们可以储存当前标准窗口的内容后再退出
非标准窗口转储
使用以下的两个函数:
滚屏操作
在行缓冲模式中,若输出的内容超过了终端的 bottom line,将会自动滚屏 (旧的输出将会向上滚动,为新的输出留位置)
我以为 ncurses 中的窗口默认滚屏,结果当输出超出屏幕时,反而向右溢出了
int scrollok(NIN*, bool); // 在 NIN 指针指向的第四中并启(tras)/浸闭(false) 浸厚 satscrapg(int x, int line); // 在 scrolloy 开启机 可以在面口中位之一个消屏机 在他区域内有消屏 // 消屏区从第(x) 行开池,凡在 [line) 行
颜色设置
有时候我们希望改变窗口的背景与文本颜色, ncurses 库对此也提供了支持
在使用之前,我们先初始化颜色设置
bool has_color(void); // 返回该环境是否支持颜色设置
int start_color(void); // 初始化颜色设置
在 start_color() 成功调用后,一系列的常量将会产生,例 COLOR (支持的颜色数量),COLOR_BLACK (原色),COLOR_WHITE...
我们使用 init_pair() 函数来领建 背景·文本的颜色对,并用 attron() 函数激活
接下来,直到 attroff() 关闭之前,所有输出的 背景·文本 都尽是指定的颜色对
其他输出文本效果
attron() 与 attroff() 不仅可以用来设定颜色,还能够实现许多输出文本效果
这些效果通过一系列常量来代表: A_BLINK, A_DIM, A_UNDERLINE
例如: attron(A BLINK). 那么在 attronoff(A BLINK) 之前输出的文本都将闪烁显示
这些效果的复合 (甚至与颜色的复合) 可以通过常量间的或 | 运算简单的实现 (这与 win
例如 attron(A_BLINK | A_UNDERLINE) 代表文本闪烁且加下划线显示
这里附上 ncurses 提供的所有输出文本效果常量
Reference

    Debian nourses manpage
    適開 scrollok

    窗口转链 scr_dump

  2. <u>tiga-Unix/Linux下的Curses座开发指面——第三章curses库面口</u>(这好像也是转载的,但是翻译的很好,也很全)
 3. KeBlog-ncurses (Kewth NB)
  4. <u>ztq-ncurses库 常用诱数及基本使用</u> (有一个通过子曾口实现分屏的程序,gp 的时候可以借鉴,修改一下参数)
 5. ncurses 输出修饰 (包含了 ncurses 提供的一系列 attr 前缀函数,对输出进行修饰,很有用)
                           好文要顶 关注我 收藏该文 微信分享
② □型製目天下第一
粉丝 - 10 美注 - 5
                                                                        4 0
①推荐 り反对
                                                                             升级成为会员
« 上一篇: ENGG1310 P3.2 Resistor, Inductor, Capacitor, and RLC Circuits
» 下一篇: MATH1851 Laplace Transforms
 posted ⊕ 2023-04-14 14.55 四季夏目天下第一 阅读(8083) 评(8(8) 編輯 敬蔵 孝樹
9. 登录后才能查看或发表评论,立即 登录 或者 迅迅 博客园首页
₫ 豆包
 快速输出 提效神奇
 剛練行:
- (HelloGitHub) 第 105 間
- 句话. 我让 Al 腎務飲了个 P 閻网站!
- 句话. 我让 Al 腎務飲了个 P 閻网站!
- 记一次、NET某工业模型软件 崩溃分析
- Docket慢像投資。
- Docket慢像投資。
```

