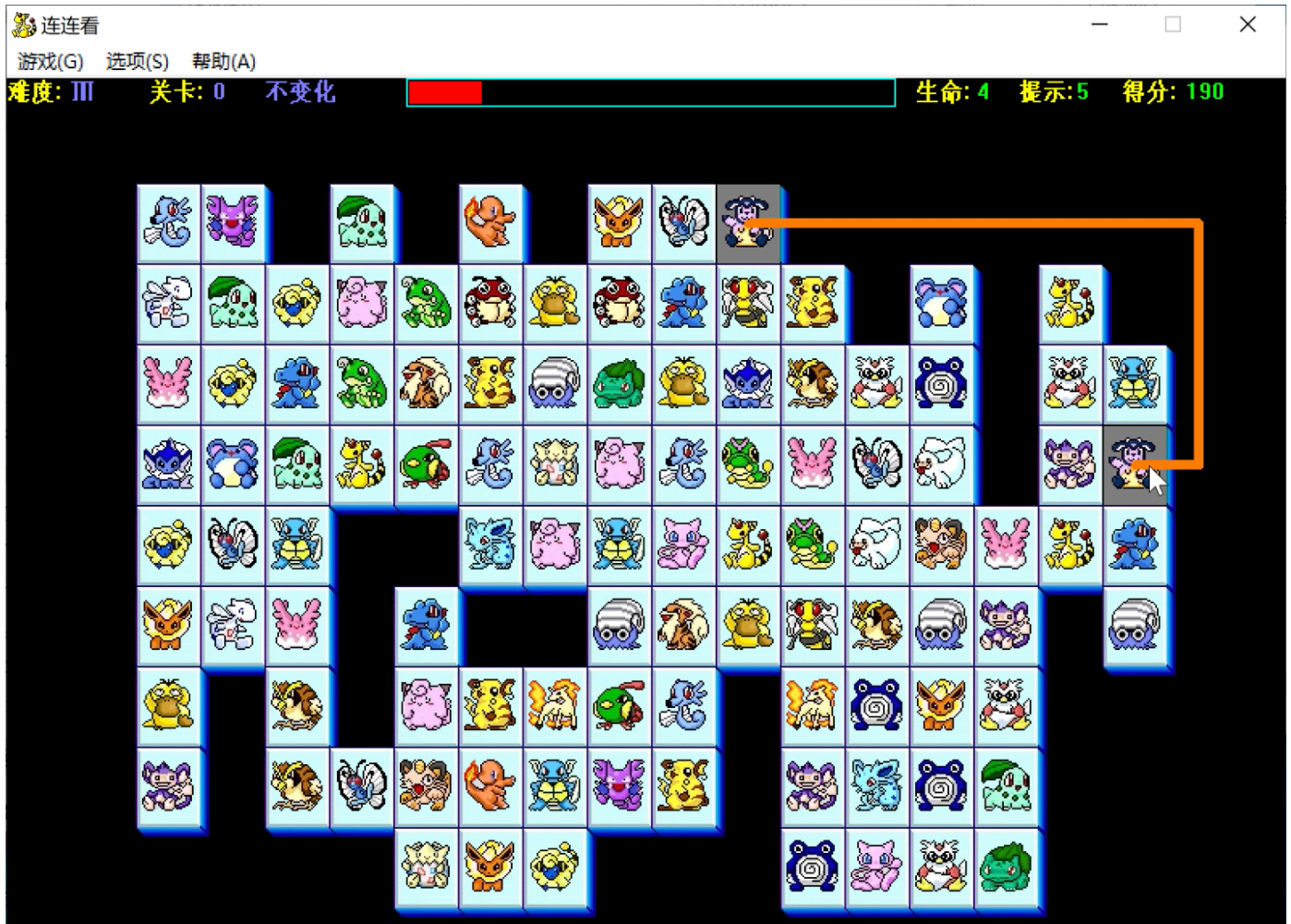


Project 1: QLink

连连看



QT GUI 基础，从 AnalogClock 到推箱子

GUI 编程的大体思路

大多数 GUI 编程，是基于事件的（Event-driven）。在这种事件驱动的编程模型中，存在事件和处理函数两个概念：

- 事件（Event），如鼠标移动、鼠标点击、键盘输入、窗口大小改变等，用户也可以自己定义事件。
- 事件处理函数（Event Handler），程序员可以将函数与事件进行绑定，当某个事件触发之后，系统会自动调用对应的处理函数。

简单来说，GUI 程序的大体思路是这样的：

```
int main()
{
    // 准备一些基本的界面
    prepare_display();
}
```

```

// 为事件绑定处理函数，这部分往往和准备界面是混在一起的
// 这里单独提出来，主要是为了突出事件和处理函数的重要性
bind_handlers_to_events();

// 开始进入主循环，只要程序不退出，该循环不会退出
for (;;) {
    // 等待一个事件发生
    event = wait_for_event();
    // 获取绑定在该事件上的处理函数
    handlers = get_handlers(event);
    // 调用这些绑定在事件上的处理函数
    for (handler in handlers) {
        handler();
    }
}
return 0;
}

```

因此，我们要实现一个 GUI 程序，重要的是准备界面和写处理函数。

QT 中的事件和处理函数

在 QT 中，有 Signal、Slot 和 `connect` 的概念，其中 Signal 和 Slot 对应着前面的 Event 和 Event Handler，`connect` 则将两者进行绑定。

```

// 一个 QLabel 实例，类似于文本框，但是只用来显示文字，用户不能修改其中的内容
QLabel *label = new QLabel;
// 一个滚动条实例
QScrollBar *scrollBar = new QScrollBar;
// 下面的这个 connect 增加了这条绑定关系：
// 当这个滚动条上发生了 valueChanged(int) 这个事件的时候（也就是滚动条被拖动的时候），
// label 对象上的 setNum(int) 函数被调用
// 最后的效果，是用户拖动滚动条，label 里面会动态显示当前滚动条的位置（一个数字）
QObject::connect(scrollBar, SIGNAL(valueChanged(int)),
                 label, SLOT(setNum(int)));

```

Practice 1

请阅读给出的代码和注释，将其运行起来

Practice 2：增加秒针

1. 将分针的颜色改为蓝色
 2. 增加秒针，颜色为红色，长度大于分针长度，形状为任意四边形
- 请注意，你的程序是写给用户的，因此不要只为了完成作业而脱离实际，请考虑让用户能看懂这个是秒针

Practice 3：时间暂停

1. 增加按键功能，按下 P 键则钟表的时间停止

Qt 中的按键也是一个事件，由于非常常用，Qt 已经帮我们做好了处理函数：

```
[virtual protected] void QWidget::keyPressEvent(QKeyEvent *event)
```

我们的 AnalogClock 类是 QWidget 的子类，因此可以 override 该函数，来处理键盘按键事件。具体怎么用，怎么获取按下的是哪个键，还需要你去看一下文档哟。

2. 再按 P 键恢复

- 此处的实现不需要考虑时间连续性，因此在恢复时表针会发生跳跃，显示真正的当前时间
- 思考：如何模拟时间暂停，按 P 键恢复之后，时间从暂停时的下一秒开始？

Practice 4：移动的人物角色

1. 在屏幕上，画一个圆圈表示人物角色
 2. 按键 WASD 控制该圆圈上下左右移动
 3. 当人物在边界走时，可以从另外另外一端的边界出来
- 思考：如何让用户自动向前走，WASD 只是控制自动移动方向？

Practice 5：推箱子

1. 在 4 基础上，再画一个方块表示箱子
2. 当人物移动到箱子旁边，并向箱子方向再次移动时，会推动箱子一起移动

通过上面的练习，你应该已经掌握基本的 QtGUI 编程的能力。一些没有涉及到的操作和用法，可以在此后的实践中进行学习，请善用 QT 的帮助文档和搜索引擎。

QLink

在连连看游戏中，会有一个地图，地图上有许多不同种类的方块，通过将相同种类的两个方块相连，可以将这两个方块消除，用户获得分数。

在整个连连看的过程中，除了处理用户的操作之外，还有几个比较特殊的部分特别需要注意：

- 随机地图的生成
- 判断两个方块是否可以通过两次以内的折线进行连接
- 判断剩余方块是否还有解

除此之外，具体的功能要求如下：

RPG 机制

不同于传统的连连看，我们的 QLink 使用 **RPG 模式**进行，即玩家需要控制一个角色在地图的空地上移动（角色显示可自行选择）。

- 激活：当角色处于方块旁且再次向方块方向移动，会激活该方块（请使用某种方式表示该方块被激活）。
- 消除：如果此次激活的方块和上次激活的方块是同种类，且可以通过两次以内的折线连接，则该两个方块被消除，玩家获得分数。（请绘制出将两个方块连接在一起的折线）否则，上次激活的方块被自动变为未激活状态，换句话说，每个角色在地图中只有 0 个（游戏刚开始时，或刚刚消除完一对方块时）或者 1 个激活的方块。

计分

不同种类的方块可以有不同的分值，具体规则可以自行制定。界面中应时刻显示玩家的分数。

倒计时和游戏结束

有两个情况可以导致游戏结束：

1. 倒计时结束；
2. 没有可消除的方块对（所有方块均被消除也属于这一种）。

界面中应时刻显示游戏的倒计时。

开始菜单

至少包括以下按钮：

- 开始新游戏
 - 可选择游戏模式：单人模式、双人模式（具体看后文）
- 载入游戏
- 退出游戏

单人模式

游戏开始时，会随机生成地图，并随机玩家角色位置。随后玩家可控制角色移动，以激活和消除方块。

道具

道具通过随机方式出现在地图的空地上，当角色与道具出现在同一位置时，该角色触发道具效果，道具消失。

- +1s：延长剩余时间 30s
- Shuffle：所有方块位置重排
- Hint：10s 内会高亮一对可能链接的方块，被消除后会高亮下一对，直到 10s 时间结束
- Flash：5s 内允许通过鼠标单击移动角色位置，角色移动到的位置必须通过空地可到达，否则点击不产生任何效果。如果点击到方块，且角色可以移动到该方块旁，则角色移动到该方块旁，且该方块被激活。如果方块四周有多个位置可以让角色停留，则角色移动到其中任何一个位置均可。

双人模式

两个玩家的两个角色在相同的地图上进行游戏，以结束游戏时双方的分数决定谁为赢家。

道具

在单人模式的基础上，增加：

- Freeze：对手 3s 内无法移动
- Dizzy：对手 10s 内移动方向颠倒（上下左右颠倒）

此外，

- Hint 道具的效果对两个玩家均可见；
- +1s 道具的效果对两个玩家均有效。
- 双人模式下，没有 Flash 道具

暂停和存档

- 在暂停时，可以保存游戏（Save）和载入游戏（Load）
- 保存游戏会将当前游戏的所有状态以任意格式保存到磁盘上的文件
- 载入游戏时，读取文件，并从中恢复状态