

Final Report

b0g队

3200105948 丁静怡

3200105453 曹田雨

Ch1 项目介绍

这是一个简易的迷宫游戏。游戏的背景是大家非常熟悉的**48h核酸检测**。

游戏开始时，玩家会获得一个简单的示意图——注意，它不是实际的迷宫，只是标注出了每个地点的大致方位（实际上，这正是我们所在的紫金港校区的示意图）。同时，机器人小冰（Little Ice）会告诉你现在所处的地点（区域）。

在示意图中显示的九个地点中，只有两个地点是**可能存在**核酸检测点的，而只有其中一个地点**真正存在**检测点。因此你需要即刻出发寻找检测点，以守住宝贵的48小时核酸证明。每走一步，贴心的小冰都会告诉你当下所处的地点以及此时可以前进的方向，输入东（E）西（W）南（S）北（N）即可朝相应方向前进一步。注意，因为是48小时核酸，你必须在**48步之内**找到核酸检测点，否则就算游戏失败。同时，为了防止有人嫌麻烦不想做核酸，不利于疫情的常态化防控，我们偷偷准备了“**作弊模式**”：当输入“help”时，就可以得到**完整地图**！

除了上述的常规模式（primary mode），我们还为**闲得无聊**的玩家准备了“**地狱模式**”（inferno mode）：对应一个大得多的迷宫，以及高达1000的步数上限。~~（其实只是写迷宫的人写完发现太难了，但又不想浪费罢了）~~

Ch2 具体实现

2.1 map 类

在实现map类之前，我们首先需要定义一个struct Pos：

```
struct Pos{
    int x;
    int y;
    int mode;
    int loc;
    Pos(int m, int px=0, int py=0);
    bool operator==(const Pos& that);
    bool operator!=(const Pos& that);
    inline int UpdateLoc();
};
```

- Pos中存放的是一个坐标点的横纵坐标以及对应地点（区域）；同时，模式信息（mode）也需存放在其中；
- 为了方便后续对位置的判断，我们在Pos中重载了 == 和 != 两个运算符；
- UpdateLoc() 函数的作用是更新坐标点对应的地点。

map类的作用是存放地图的相关信息，包括每个坐标点的通断、出口和入口的位置等；并实现地图的生成、打印等操作。

```
class Map{
private:
    int nrows; // number of rows
    int ncols; // number of columns
    int **node; // state of node
    Pos pin; //position of entrance
    Pos pout; //position of exit
    int mode; // mode 0(primary), 1(inferno)
    int ROWNUM;
    int ROWNUM_UNIT;
public:
    map<int,string> dict;
    Map(int mode);
    virtual ~Map();
    virtual void SetExit();
    virtual void SetEntrance();
    virtual void dig(int px, int py);
    virtual void initMap(); //create map
    virtual void printMap(Pos p); //print map
    virtual Pos GetE(); //get the position of entrance
    virtual Pos GetX(); //get the position of exit
    virtual bool isBlock(Pos p); // return true for wall
};
```

其中值得一提的是地图的生成~~（是作者坑害玩家的得意之作）~~。

起初，我们试图通过随机数决定每个坐标点的通断，再用DFS判断生成的地图能否走通。但经过多次尝试，我们发现，这种方法效率低且生成的地图较为简单，可玩性低。

因此，我们改进了地图的生成方式，通过DFS算法直接生成地图。具体思路如下：

- 最开始设置地图内所有点均为Wall（即走不通），并随意选取其中一点作为起点（这里使用随机数生成）；
- 从起点开始，把自己想象成一只“地鼠”向四周挖隧道，每次随机选择一个方向挖，直至挖不动为止；（具体代码见 dig(int px, int py) 函数）

用这种方式生成的迷宫，任意两点之间一定有且仅有一条正确的道路，大大提高了可玩性。

2.2 player类

player类的主要作用是记录玩家的位置信息，并实现玩家的移动等操作。

```
class Player{
private:
    Pos temp; // current pos
    Pos x; // exit of the maze
public:
    Player(Pos e, Pos x): temp(e), x(x) { }
    void move(string d); // move one step to d(N,E,S,W)
    void play(Map, int); // play in the map m
};
```

其中最重要的是 `void play(Map, int)` 函数，主要实现以下功能：

- 提示玩家当前可移动方向；
- 根据玩家指令改变当前位置坐标 temp；
- 判断是否进入新区域，如果是则提示进入新区域；
- help 作弊模式：支持地图提示；
- 提示玩家剩余步数，判断游戏结果。

2.3 其他

我们通过使用"windows.h"库, 对不同提示语采用不同颜色显示, 游戏界面更加美观。

Ch3 测试结果

- 游戏开始：
 - 给出紫金港核酸检测简版地图
 - 选择游戏模式（常规模式或地狱模式）

```

Little Ice: Welcome to ZJU Nucleic Acid Testing Challenge! ! !
Little Ice: Here is the campus map.
Little Ice: 'x' represents the nucleic-acid test available spot.
+-----+-----+
|               |               |               |
| Lantian       |               |               |
| Hall          |               |               |
|               | GangWan      |               |
|               | Community    |               |
|               |               |               |
|               |               | Swimming   |
|               |               | Pool        |
|               |               |               |
+-----+-----+-----+-----+
|               |               | ~~~~|     | |
|               |               | ~~~~|     | East  |
| QiuShi         | West        | ~|     | Building(x)|
| Lecture        | Building    | ~|     |
| Hall           |             | ~|     |
|               |             | ~~~~|     |
|               |             | ~~~~~|     |
|               | Graduate     |               |
|               | School       | Agriculture |
| Yuhu           | Building(x) | & Medicine |
| Hall           |             | Library      |
|               |             |               |
+-----+-----+-----+-----+

```

Little Ice: You can choose primary mode(0) or inferno mode(1): ☐

- 游戏过程:
 - 游戏开始，提示起点所在区域及核酸检测点所在区域
 - 提示help模式
 - 每步提示可行进方向(N, E, S, W)

本项目是对本学期OOP课程内容的简单应用，体现了面向对象编程的思想，完成了一个简单但好玩（写地图的人说的）的迷宫游戏。同时，游戏背景结合时事，相信大家都会狠狠共情！