

说明文档

目录

说明文档	1
一、 小组成员	1
二、 基本功能完成情况	1
1. 绘制模块☑	1
2. 执行模块☑	1
3. 交互模块☑	2
三、 游戏基本操作说明	3
四、 分工情况	4
1. 刘雅迪	4
2. 田睿	4
3. 薛雨泉	4

一、小组成员

组长：刘雅迪，2021010521，能动 14

组员：田睿，2022012200，新雅 23

组员：薛雨泉，2021010363，机械 10

二、基本功能完成情况

1. 绘制模块☑

详细代码见 `draw.h` 文件。

主要实现 `auto_save` 函数来绘制保存地图，其中调用 `save` 函数来保存地图状态，而 `save` 函数中调用 `draw` 函数来绘制地图、`draw_robot` 函数来绘制机器人。

灰色区域表示没有灯的地图格，蓝色区域表示有灯未点亮的地图格，黄色区域表示有灯且被点亮的地图格。

2. 执行模块☑

详细代码见 `Operator.h` 文件。

主要实现对外接口 `robot_run` 函数，该函数读入文件中的指令序列并实行相应操作，最后判断并输出结果。

读入指令序列：`readin` 函数；执行指令序列：`operate` 函数。

由于读入主函数序列时需要读入 Pn 操作，故 `readin` 函数使用多态。执行函数 `operate` 中使用 `call` 函数来调用终端，封装基本的动作并调用 `draw.h` 中 `auto_save` 函数保存。

3. 交互模块☑

详细代码见 `main.cpp` 文件。

3.1 加载地图☑

地图存放在 `maps` 文件夹中，测试地图文件名为 `map.txt`。

当玩家在命令行中输入“LOAD maps/map.txt”时，程序会自动读入 `maps/map.txt` 中的内容。

3.2 设置自动保存☑

调用 `draw.h` 中的 `auto_save` 函数。

当玩家在命令行中输入“AUTOSAVE OFF”时，程序被设置为不保存游戏过程图；令保存的游戏过程图存放在 `imgs` 文件夹中，当玩家在命令行中输入“AUTOSAVE imgs/%4d.bmp”时，游戏过程图会自动保存为 `imgs/0000.bmp` 等。如：

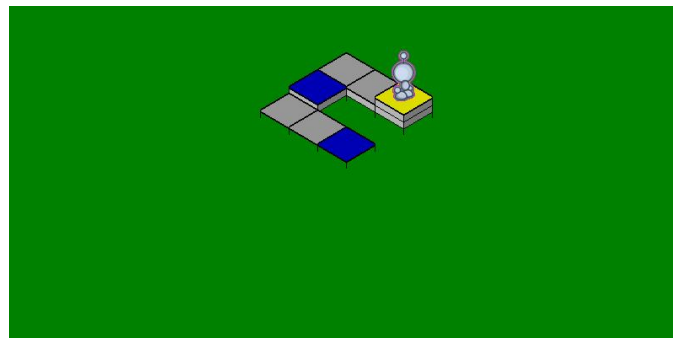


图 1

3.3 设置步数限制☑

更改执行指令序列的步数上限，即 `game.limit` 的值。

3.4 输出配置

当玩家在命令行中输入“STATUS”时，程序会调用 `printmap` 函数，输出当前配置。如：

```
STATUS
Map Name: maps/map.txt
Autosave:
Step Limit: 100
223
2
111
Robot is facing left.
Proc Limit:
3 4 4 4
```

图 2

3.5 新建指令序列

当玩家在命令行中输入“OP ops/xxx”，然后输入正确格式的指令序列，指令序列会被保存为 ops/xxx。若输入的指令序列中使用的过程数或任一过程的指令数超出了地图指定的上限，程序会输出“limit exceeded”来提示错误。

3.6 执行指令序列

调用 `Operator.h` 中的 `robot_run` 函数，如当玩家输入“RUN ops/ops.txt”时，程序会执行 ops/ops.txt 中储存的指令序列并输出执行结果。若正在执行的指令不适合当前机器人所在单元格，如在地图边缘“碰壁”，程序会输出类似“WARNING”的提醒。

3.7 结束程序

当玩家在命令行中输入“EXIT”时，程序会输出“You are exiting the game”并退出循环结束程序。

三、游戏基本操作说明

当程序运行时，玩家可依次输入：

LOAD <MAP_PATH> 加载地图

AUTOSAVE {<SAVE_PATH>|OFF} 设置是否自动保存

RUN <OP_PATH> 执行指令序列

EXIT 结束游戏

除此之外，玩家可在上述输入中间输入“LIMIT <OPS>”来重新设置步数上限；输入“STATUS”查看当前地图情况；输入“OP <OP_PATH>”与对应指令序列来创建新指令序列文件。

四、分工情况

1. 刘雅迪

- 协调分工情况，组织组员定期交流进度以及安排后续任务
- 编写“交互模块”代码
- 整合三部分代码并进行调试
- 撰写说明文档内容
- 制作演示视频

2. 田睿

- 编写“绘制模块”代码
- 帮助调试代码

3. 薛雨泉

- 编写“执行模块”代码