

Box Mover 设计文档

Java 程序设计 Project1

刘雅诺 张彭景 倪敏悦

三． 项目要求

3.1 基本功能（必须实现）：

- 1) 在命令行界面上正确显示 TA 提供的所有 9 幅地图，并能正常开始游戏。地图应在响应用户事件后进行重绘。地图的显示方法可以自定义；
- 2) 正确响应玩家的键盘事件。用户通过输入转向指令控制机器人在地图上将箱子推向相应的地点。以下指令为必须实现的基本指令：左转、右转、前进、后退、悔 1 步棋、选择关卡、重新开始、帮助、退出。同时，程序应具有一定的容错性，即当用户输入非法指令时，程序应能给出合法指令提示并继续游戏；
- 3) 程序应具备一定的游戏结果判断能力。这里可以分为 2 种情况：
 - a) 当所有箱子都已推到目标地点时，应自动提示用户游戏胜利；
 - b) 当所有箱子都已不再可以被推动，即游戏进入死局时，应自动提示用户闯关失败，退出本局游戏。以下图为例，图 2-1 中红色箱子仍然能被推动，则游戏可以继续进行；而图 2-2 中，游戏已陷入僵局，可以继续推动的箱子个数为 0。



图 2-1



图 2-2

鉴于死局的逻辑判定比较复杂，我们将对不同程度的判断给出相应的得分。

A 类死局定义为：机器人可以自由走动，而所有箱子都已被推到角落，即如图 3-1 所示。完成第一类死局判断，得 5 分。

B 类死局定义为：机器人活动范围被局限，有至少 1 个箱子无法到达和推动，如图 3-2 所示，机器人被局限在棋盘的下半部分区域，三个红色框内箱子均不能到达。注意：在图 3-2 的例子中棋盘被划分为 2 个区域，而不同的地图可能会在游戏过程中被划分为多个区域，机器人可能会有多于一个区域不可达。完整实现第二类死局判断得 10 分。



图 3-1



图 3-2

4) 文档及代码要求

- a) 设计开发文档包含项目构思、实现过程等相关内容，参考附件[设计文档.pdf](#)；
- b) 用户手册则是面向用户的文档，具体描述游戏玩法等，类似产品说明书，可参考[用户手册.pdf](#)；
- c) 代码应具有良好的代码风格，包括代码格式、注释以及类的封装，请尽量避免将所有代码写在同一个类文件中。

3.2 进阶功能 (Bonus)

进阶功能总分 10 分，同学们可根据个人情况选择完成下列进阶功能中的 0-2 个，TA 根据相应得分点给分。所有进阶功能的实现都需在设计文档中进行说明。

- 1) 重写 TA 提供的读写文件部分代码。不使用 TA 的 API，自主实现关卡信息的读取和保存功能，给出相应的加分；
- 2) 提供悔多步棋功能。功能可通过多次输入悔棋指令或直接输入悔棋步数进行悔棋操作，例如：用户输入两次 B 或者直接输入 B 2 均表示悔 2 步棋，TA 将根据实现过程给出相应加分；直接输入悔棋步数也需进行初始状态判断。
- 3) 自主创新。可对 TA 提供的地图进行修饰改造，也可设计奖励手段为用户提供一些道具，例如：炸药、木板等，方便用户在陷入死局时另辟蹊径。注意：当用户拥有炸药、木板等道具时，游戏的结果判断应作相应修改。

四． 项目提交

5.1 Deadline : 2013-11-17 (每延期一天扣分 20% , 亦即迟交 5 天得 0 分)

5.2 提交文件 :

- 1) 所有源代码
- 2) 开发设计文档
- 3) 用户手册

5.3 检查方式 :

- 1) 在规定 Lab 课时间段内 , 学生向 TA 进行现场演示
- 2) TA 会在检查完毕后给出相应的成绩
- 3) 需要查分的同学可联系 TA 了解详细得分情况
- 4) 严禁抄袭 , 一经发现 , 以 0 分处理

5.4 上传方式 :

将所有需上传的文件以压缩文件格式上传以下地址 , 文件命名格式为 “学号_姓名” :
ftp://10.132.141.33/classes/13/121 程序设计 (陈荣华) /WORK_UPLOAD/Project1

五． 评分标准

基本功能 (100)	功能实现	调用 TA 提供 API 正确显示并重绘地图		10	
		正确响应 用户指令	左转、右转、前进、后退	20	
			悔一步棋	5	
			选择关卡、重新开始	5	
			帮助、退出	5	
		结果判定	赢局判断	10	
			死局判断	A 类死局判断	5
				B 类死局判断	10
	程序容错性（非法指令的处理等）		10		
	文档及 代码风格	项目开发设计文档		10	
用户手册		5			
良好的代码风格		5			
附加功能 (10)	附加功能	重写 TA 提供 API 中 IO 部分代码		5	
		根据用户指定步数悔棋		5	
		自主创新		5	

六. 程序运行流程

游戏规则：在设定的游戏地图里，箱子只可以推，不可以拉，而且玩家一次只能推动一个。

胜利条件：把所有的箱子都推到目的地。

游戏流程：

1. 游戏开始，读第一张地图，正确显示地图。其中■代表墙，□代表箱子，☆代表玩家，○代表目的地，●代表被箱子填满的目的地。（如图 1）
2. 使用 LRUD 分别代表向左、右、上、下移动，在前进方向上没有墙阻挡时，可以一次移动玩家（☆）一格；当玩家与箱子（□或者●）相连接时，如果对应方向没有墙的阻挡，则可以将箱子推动一格。
3. 当玩家将箱子推入到目的地○之后，○变成●，当所有的○变成●之后，在屏幕提示本局游戏闯关成功，要是有一局直接提示进入下一局并显示游戏界面。



图 4 游戏结束并继续闯关

4. 当玩家每一个箱子都无法推动陷入死局（注：此处死局参见本文档对死局的具体定义，与普通意义死局不同），提示玩家。



图 5 游戏出现死局

5. 游戏输入端应该具有一定的容错性，能够提示非法输入并继续输入。

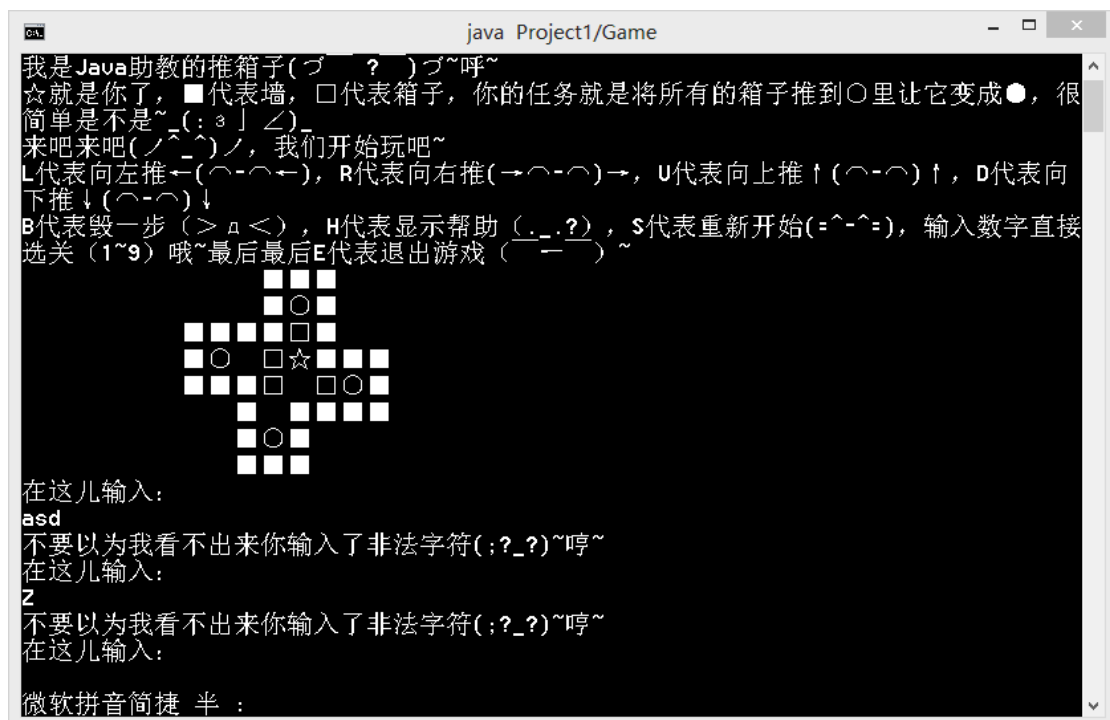


图 6 非法指令提示

6. 在输入选项可以选择重新开始。



图 7 游戏重新开始

7. 在输入选项可以选择帮助指令。

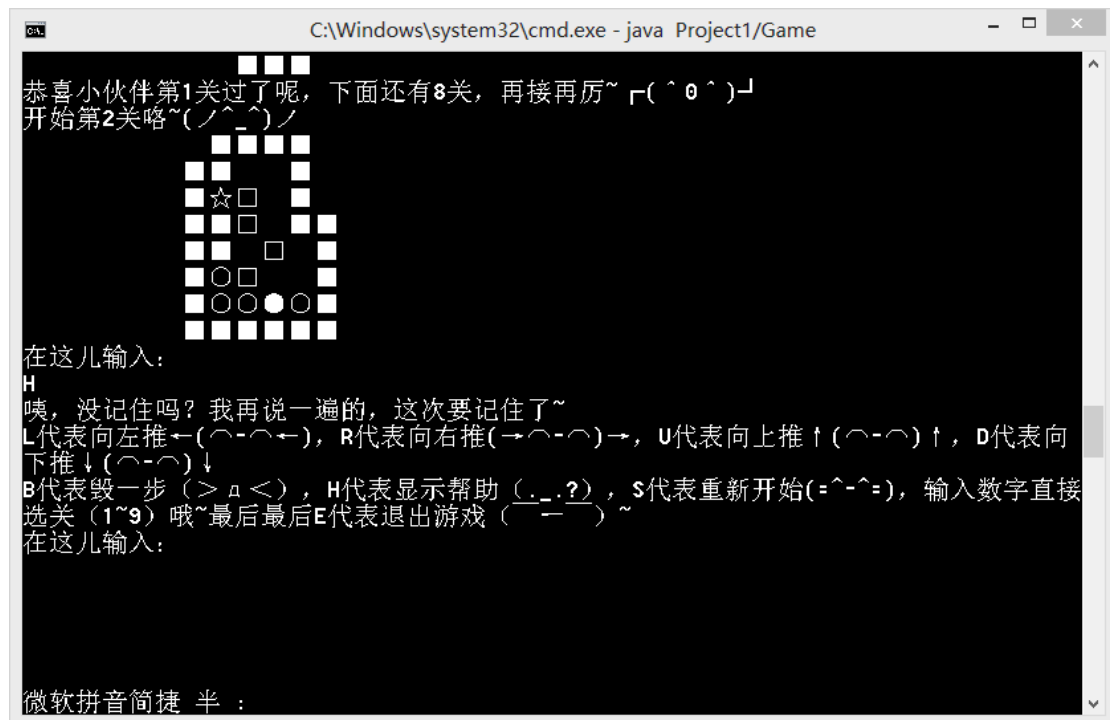


图 8 帮助指令

8. 在输入选项可以选择输入数字选择关卡, 需对数字是否在 1~9 进行判断。



图 9 选择关卡

9. 在输入选项可以选择悔一步，要是多次悔步至初始状态需提示。

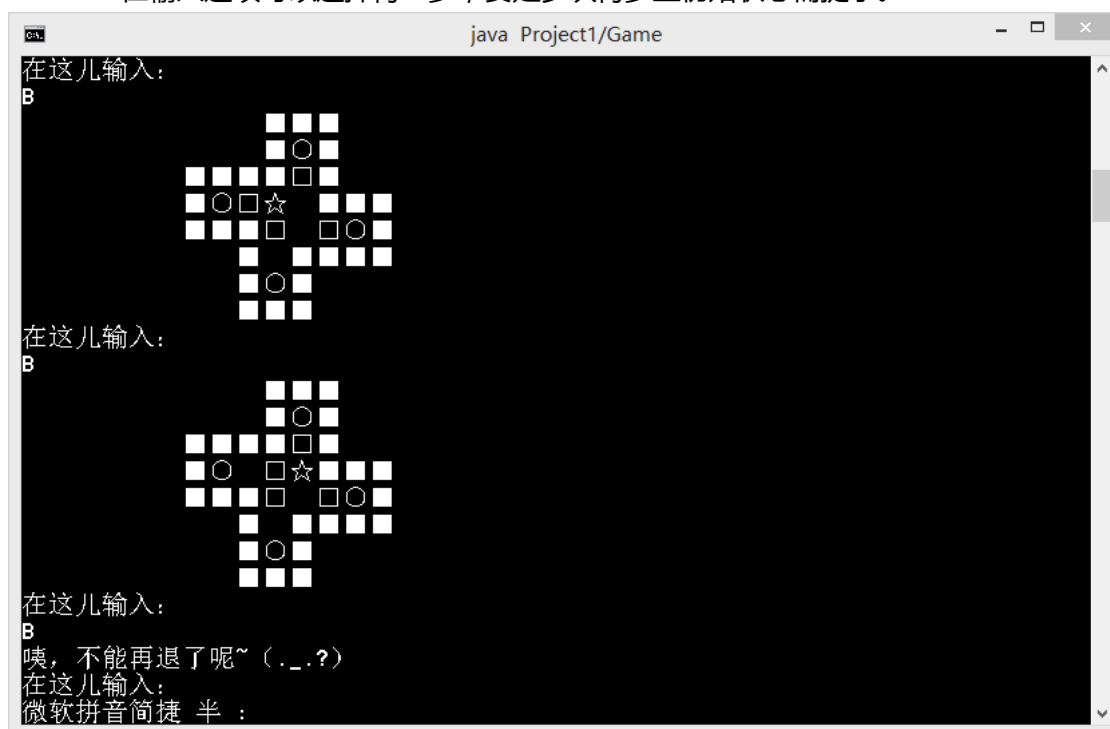


图 10 多步悔棋