

程序设计基础大作业

0. 大作业方案

- (1) 大作业满分 10 分，占课程总成绩 10%。
- (2) 含源程序提交（在课程网站-图文作业“大作业”中提交）和验收（在机房验收）两个环节，验收时每人演示 2 分钟，回答问题或报告 2 分钟。为了清楚解释你的实现，你需要进行合理注释，甚至撰写书面报告。
- (3) 可在以下 1 和 2 中任选一项加以实现。鼓励在所选项目中做创新增加功能。建议大家量力而行，不鼓励内卷，仅鼓励感兴趣的同学勇敢迎接挑战。
- (4) 大作业将实行更为严格的作弊惩罚机制，一经发现存在全部或部分代写、拷贝、照着写、…等作弊行为，即倒扣 10 分。
- (5) 不得转发或外传本次大作业。
- (6) 本次大作业的最终解释权归教学团队所有。非常欢迎大家提出疑问和建议。

1. 2048 游戏程序

1.1 作业目标

实现一个 2048 游戏程序，综合训练包括程序流程控制、模块设计、基本操作表达、简单数据的描述，以及复杂数据的描述与操作等过程式程序设计能力，以及程序编写与调试能力。

```
0      0      0      0
2      0      2      0
16     0      0      0
2      8      0      0

your step is:11
```

图 1 2048 游戏示意图

1.2 游戏规则

玩家要想办法在 4×4 棋盘的 16 格范围内凑出一个整数“2048”，具体包括：

- (1) 在一个 4×4 的棋盘上，随机初始化两个方格的值为 2，其中一个方格应在棋盘四个角的某一个位置，另一个方格位置随机，其它方格的值置为 0。
- (2) 玩家每次可以选择上下左右一个方向，每选一个方向，所有的数字往所选择的方向移动一次，相同的数字在相遇时会相加。

- (3) 一次移动结束后，系统同时会在 0 位置随机出现一个 2 或 4，2 出现的概率应比 4 高。
- (4) 凑出"2048"方格，或没有 0 位置且没有相邻数字相同时，游戏结束。
- (5) 游戏计分标准：初始得分为 0，每选择一次方向减 1 分，每次合并数字都会增加对应数字的分数，例如，两个 4 合并在一起则会增加 4 分。如果最终凑出"2048"方格则视为通关，没有凑出"2048"方格且游戏结束视为失败。

1.3 任务要求

- (1) 使用 C 或 C++ 语言编写可编译执行的 2048 游戏源程序。
- (2) 调试程序，确保玩家能正常、完整地玩游戏，游戏机制符合游戏规则要求，游戏带有方向控制、计分等基本功能。
- (3) 设计并实现 至少两个不同难度的关卡或 至少一个趣味性功能。
- (4) 实现简单的图形化游戏界面，以及游戏介绍、游戏开始、游戏结束等展示界面。

1.4 成绩评定

- I) 6 分：完成 1.3 任务要求 (1) 和 (2)，并演示程序、回答提问。
- II) 7 分：在 I) 的基础上，确保游戏过程流畅、代码规范、注释合理。
- III) 8 分：在 II) 的基础上，完成任务要求 (3)。
- IV) 9 分：在 III) 的基础上，完成任务要求 (4)，并撰写书面报告。
- V) 10 分：在 IV) 的基础上，进行界面、算法优化和口头报告，界面友好、有趣，口头报告生动。

2. 攒蛋托管机器人

2.1 作业目标

实现一个攒蛋托管机器人，综合训练包括程序流程控制、程序分解与复合、多模块程序组织、基本操作表达、简单数据的描述，以及复杂数据的描述与操作等过程式程序设计能力，以及程序编写与调试能力（用 C/C++ 语言）。



图 2 掇蛋托管机器人示意图

2.2 游戏背景

“掇蛋”，是一种在苏北地区广为流传的扑克游戏，是由地方的扑克牌局“跑的快”和“八十分”发展演化而来。牌局采用四人结对竞赛，输赢升级的方式进行。使牌局更富有娱乐性和刺激性。

这里采用江苏出台的《江苏省掇蛋竞赛简易规则》（2017 版）进行介绍。为简化问题，删减了部分内容，因此作业中遵循的规则请完全按照下面的“2.3 规则”。例如：本次大作业中需要考虑级牌等，但不考虑逢人配、抓牌顺序、升级、贡牌、还牌等规则。如果以下内容让你难以理解，试着开始一局掇蛋游戏吧！

2.3 游戏规则

2.3.1 牌张

掇蛋采用 2 副扑克牌，分为黑桃♠、红心♥、方块♦、梅花♣四种花色牌和小王、大王两种王牌。全副牌共有每种花色牌各 26 张（2、3、4、5、6、7、8、9、10、J、Q、K、A，各 2 张），小王、大王各 2 张，共 108 张。

全副牌被一桌四位牌手按序抓完，每位牌手应持有的全部牌张为 27 张，称为全手牌。

2.3.2 牌手

一桌四位牌手两人搭档各为一方。牌手本人为本家，位于对面的同伴为对家，位于左方的对手为上家，位于右方的对手为下家。本家和对家为己方，上家和下家为对方。

一副牌中，按牌手出完各自全手牌的先后，称为上游、二游、三游、下游。

2.3.3 级数

从 2 至 A 的从小到大依次排列的每一个序数，包括 2、3、4、5、6、7、8、9、10、J、Q、K、A 共十三个级数。

2.3.4 级牌

牌点与上副牌上游所升至级数相同的所有花色牌为级牌，首副牌的级牌为 2。本题中，当前轮次的级牌由输入给出。

2.3.5 牌型

1. 单张：任意一张牌。例如图 1 中的 J
2. 对子：两张牌点相同的牌，包括两张大王和两张小王。例如图 1 中的 66 和大王大王
3. 三连对：三个牌点相邻对子。例如图 1 中的 554433
4. 三同张：三张牌点相同的牌。例如图 1 中的 222
5. 二连三（钢板）：两个牌点相邻的三同张牌。例如 444333
6. 三带二：一个三同张和一个对子。例如图 1 中的 22233
7. 顺子：五张牌点相邻的单张。例如图 1 中的 KQJ109
8. 同花顺（火箭）：花色相同的顺子。例如图 1 中的梅花 65432
9. 炸弹：四张或四张以上牌点相同的单张。例如图 1 中的 77777
10. 四大天王（天王炸）：大王、小王各两张。

2.3.6 出牌

2.3.6.1 出牌程序

出牌以逆时针为序，领出者可出任意牌型的一手牌，各家依次跟出牌点、牌型更大的一手牌进行压制（压牌），或放弃不出（过牌）。如其余各家均放弃不出，则由最后出牌者下一圈领出牌；若此时最后出牌者已出完全手牌，则由己方同伴承接下一圈领出牌（对家接风）。重复以上程序直至全副牌出完。

2.3.6.2 牌点与牌型大小

1. 牌点由大到小排列为：大王、小王、级牌、A、K、Q、J、10、9、8、7、6、5、4、3、2。注意：默认情况下，级牌可以压除大小王以外的牌，但是级牌也可以当作它的牌点使用，例如图 1 中，级牌为 10，手牌有 10 K Q J 9，那么这副牌可以当作 KQJ109 的顺子打出。
2. 单张、对子、三连对、三同张、二连三、三带二、顺子，同牌型比较牌点大小，牌型之间不能压牌。三带二仅比较三同张牌点大小。
3. 炸弹：张数多者大，张数相同比较牌点大小。炸弹可压单张、对子、三连对、三同张、二连三、三带二、顺子等牌型。超过 5 张的炸弹可以压同花顺。
4. 同花顺：比较牌点大小。同花顺可压不超过 5 张的炸弹。

5. 四大天王：可压所有牌型。

2.3.6.3 报牌

牌手在出完一手牌后，如手中剩余牌不超过 10 张，应立即将报牌卡放至本家出牌区域，并准确、清楚报出剩余张数。牌手报牌后，无论是否继续出牌，均不得再次报牌，其他牌手也不得再追问。本题用命令行输出代替报牌。

2.3.7 胜负

在本题中，能够确定上游、二游、三游、下游对应的玩家时结束比赛，并且获得上游玩家的一方 A 获得胜利。此时如果 A 方中另一玩家获得二游，则 A 方积三分；另一玩家获得三游，则 A 方积两分；另一玩家获得四游，则 A 方积一分。另一方积零分。本题不考虑积分问题。

2.4 任务要求

- (1) 使用 C 或 C++ 语言编写可编译执行的掇蛋游戏源程序。
- (2) 调试程序，确保玩家能正常、完整地玩游戏，游戏机制符合游戏规则要求，游戏带有以下基本功能：

设想你在打掇蛋。现在，上家刚刚出完牌。给定你当前手中的牌和上家刚刚出的牌，请尝试判断能否用手中的牌压过对方的牌。如果可以，输出能压过对方的最小牌型的一组牌。

“能压过对方的最小牌型”：即优先使用与对手相同的牌型，且牌面数字和花色尽量小。如果无法压过对方，再依次考虑炸弹、同花顺、四大天王。

你需要实现自动打牌功能。提供从命令行读取输入（可用于实时托管）的输入方式。

- (3) 提供从文件读取输入（可用于游戏复盘）的输入方式。
- (4) 实现简单的图形化游戏界面，以及游戏介绍、游戏开始、游戏结束等展示界面。

2.5 成绩评定

- I) 7 分：完成 2.4 任务要求 (1) 和 (2)，并演示程序、回答提问。
- II) 8 分：在 I) 的基础上，确保游戏过程流畅、代码规范、注释合理。

III) 9 分：在 II) 的基础上，完成任务要求 (3)。

IV) 10 分：在 III) 的基础上，完成任务要求 (4)，并撰写书面报告。

2.6 说明

2.6.1 从命令行读取输入

2.6.1.1 输入

1. 第一行包含 2 个数字 times 和 level，分别代表总共有几个记录和当前级牌。
2. 第二行包含 2 个数字 n 和 m，分别代表当前有 n 张手牌和对方打出 m 张牌。
3. 第三行包含 n 张牌的牌面，表示你手头有的牌，空格隔开；
4. 第四行包含 m 张牌的牌面，表示对手打出的牌，空格隔开。如果 m=0，那么第三行没有输入，代表场上没人出牌，由用户出牌。

2.6.1.2 输出

输出 times 次记录，每次输出包含至多三行如下：

第一行：首先输出“Player plays: ”。然后如果手头的牌能压过对手的牌，请从大到小输出能压过对方的最小牌型的一组牌，空格隔开。如果无法压过对手的牌，输出“Pass”。

第二行：然后输出“Player holds: ”。然后输出当前玩家手上剩下的牌。

第三行：如果玩家的手牌数量原先>10 张，在出牌后≤10 张，那么需要输出“ALERT: The number of cards in the player's hand is only x!”（x 是出牌后剩余手牌张数）。另外，如果玩家手牌数量原先>0 张，出牌后=0 张，那么默认玩家获得上游，输出“INFO: Player wins Upstream”。其它情况无需输出。

2.6.1.3 说明

1. 牌点用数字和大写字母表示，小王用“joker”表示，大王用“JOKER”表示。
2. a, b, c, d 分别代表花色的梅花♣、方块♦、黑桃♠、红心♥，每张牌的花色用半角括号括起，接在牌点后面。
3. 输入的牌面不一定有序。但是输出中，牌面必须有序。顺序应该按照牌点为第一优先级做非升序，花色为第二优先级做非降序排列（因为牌局刚开始的时候手牌是乱的☺）。
4. 在“能压过对方的最小牌型”有多种可能时（例如花色选择），随机输出一种即可。

2.6.1.4 样例 1

输入样例

1 2

16 6

JOKER 2(a) 2(b) A(c) Q(c) Q(d) Q(d) J(c) J(d) 10(a) 10(c) 9(a) 8(d) 8(d) 4(b) 4(d)

5(a) 5(a) 4(b) 4(c) 3(a) 3(a)

输出样例

Player plays: Q(c) Q(d) J(c) J(d) 10(a) 10(c)

Player holds: JOKER 2(a) 2(b) A(c) Q(d) 9(a) 8(d) 8(d) 4(b) 4(d)

ALERT: The number of cards in the player's hand is only 10!

解释

总共有 1 条输入记录，当前级牌为 2。

对第一条记录，本家有 16 张手牌，当前场上打出的牌有 6 张。本家的手牌中，能压过场上的牌（二连三/钢板 554433）的最小牌型是 QQJJ1010，其中 Q 的花色可以是 c 和 d，也可以是 d 和 d。本家打出牌后，手牌数量不超过 10 张，因此输出 ALERT 信息。

2.6.1.5 样例 2

输入样例

2 8

1 5

9(b)

9(a) 8(a) 7(a) 6(a) 5(a)

1 1

9(b)

8(a)

输出样例

Player plays: Pass

Player holds: 9(b)

Player plays: Pass

Player holds: 9(b)

解释

总共有 2 条输入记录，当前级牌为 8。

对第一条记录，本家有 1 张手牌，当前场上打出的牌有 5 张。本家的手牌中，没有能压过场上的牌（顺子 98765）的最小牌型。因此本家选择 Pass。

对第二条记录，本家有 1 张手牌，当前场上打出的牌有 1 张。本家的手牌中，没有能压过场上的牌（单张级牌 8）的最小牌型。因此本家选择 Pass。

2.6.1.6 样例 3

输入样例

1 2

16 0

joker 2(a) 2(b) A(c) Q(c) Q(d) Q(d) J(c) J(d) 10(a) 10(c) 9(a) 8(d) 8(d) 4(b) 4(d)

输出样例

Player plays: 4(b)

Player holds: joker 2(a) 2(b) A(c) Q(c) Q(d) Q(d) J(c) J(d) 10(a) 10(c) 9(a) 8(d) 8(d) 4(d)

解释

总共有 1 条输入记录，当前级牌为 2。

对第一条记录，本家有 1 张手牌，当前场上打出的牌有 0 张。本家的手牌中，能压过场上的牌（没有牌）的最小牌型是 4。因此本家选择打出 4（花色可以是 b 或 d）。

2.6.2 从文件读取输入

输入、输出格式和“从命令行读取输入”相同。唯一的区别是输入应该从.txt 格式的文件中读入。

2.6.3 自由度

你可以实现一个合理的命令行交互逻辑，用以选择命令行或者文件的方式进行输入。当然，如果你能用梅花♣、方块♦、黑桃♠、红心♥四种花色符号替换题目中的 a, b, c, d 也是很欢迎的。但是你的提示器不应该使用超出本次作业规定范围内的规则，并且不能使用随机方法，每次输出结果必须相同。

本次大作业没有 OJ 评测，你只需要保证出牌合法、剩余手牌正确等若干基本功能正确、高效即可。验收时需输入若干样例对你的程序正确性、鲁棒性、运行效率等进行检查。

2.6.4 拓展功能举例

1. “能压过对方的最小牌型”是最简单的一种出牌策略，但它不一定是最优的出牌方法。一种可能的拓展是“自己的剩余出牌次数最少的前提下能压过对方的最小牌型”，一种是“能压过对方的牌张数量最多的牌型”，还有一种是“不拆炸弹以上牌型前提下能压过对方的最小牌型”。一个较优的出牌策略可以留待有兴趣的同学继续探索。

2. 真正的掇蛋牌局，还有逢人配等规则，你也可以尝试实现这些策略，从而完成一个真实环境下的掇蛋托管机器人雏形。