

# 2021 秋季程序设计 期末 Project 文档

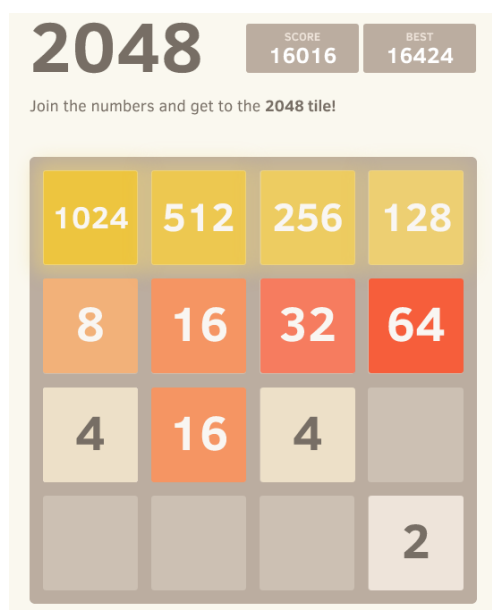
助教郭天成-20210240004

助教翟登展-21212010051

助教刘勤-19210240011

## 1 背景介绍

2048 游戏是一款单人在线和移动端游戏，由 19 岁的意大利人 Gabriele Cirulli 于 2014 年 3 月开发。游戏任务是在一个网格上滑动小方块来进行组合，直到形成一个带有有数字 2048 的方块。



该游戏使用方向键让方块整体上下左右移动。如果两个带有相同数字的方块在移动中碰撞，则它们会合并为一个方块，且所带数字变为两者之和。每次移动时，会有一个值为 2 或者 4 的新方块出现，所出现的数字都是 2 的幂次。当值为 2048 的方块出现时，游戏即胜利，该游戏因此得名。

## 2 项目目标

本项目中，同学们需要使用 C 编程语言，结合课堂知识、lab 内容，实现一个简单的、可以在 Windows 命令行下玩的 2048 游戏。期末的项目目标，将在期中项目要求上，新增内容。最后检验的是一个完整的项目。并按要求编写一份开发文档，介绍你的实现方式、开发思路、游戏特点等内容。

关于项目有任何疑问，可以在微信群、超星平台或通过邮件咨询。

## 3 期中项目基础

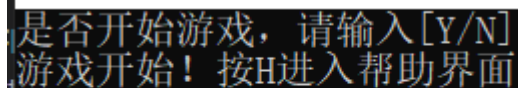
### 3.1 2048 棋盘的绘制

基础的 2048 游戏，本身是一个 4\*4 网格构成的游戏棋盘平面。

### 3.2 游戏逻辑

#### 3.2.1 开始游戏

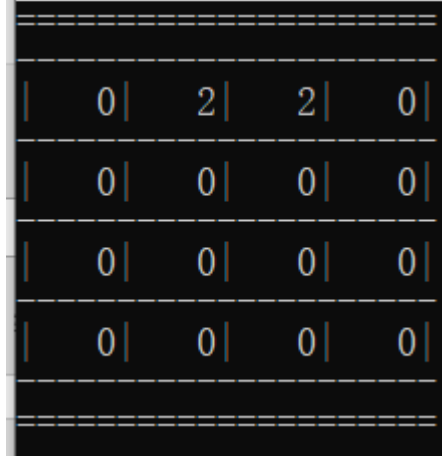
游戏启动后，进入欢迎界面，给玩家一些操作提示，并让玩家输入 Y/N 表示是否开始游戏。



```
是否开始游戏，请输入[Y/N]  
游戏开始！按H进入帮助界面
```

### 3.2.2 初始化 2048 棋盘

首先，对构建好的 2048 棋盘，向其中随机填充两个数字 2 滑块。



0	2	2	0
0	0	0	0
0	0	0	0
0	0	0	0

### 3.2.3 终端读取键盘输入，进行操作

使用 WSAD 进行操作，分别表示向上、向下、向左、向右操作。每次操作后，自动刷新棋盘，模拟滑块滑动操作，更新滑块最新的位置。同时，对于能合并的滑块，按照滑动方向进行合并，合并规则如下：

假设滑动方向为向右。滑动后，所有滑块靠右对齐。对于每一行，从左到后合并，如果相邻的两个滑块相同，则合并，否则判断之后的两个滑块。滑块不连续合并。如：

0 2 4 4 ➡ 0 0 2 8

2 2 4 4 ➡ 0 0 4 8

2 2 2 4 ➡ 0 2 4 4

合并操作结束后，随机向棋盘内，添加一个 2 或者 4 的滑块。

综上，每一次按“WSAD”键，包含了三步操作：“移动”、“合并”和“添加滑块”。

### 3.2.4 终止条件判断

当棋盘合并出 2048 滑块后，游戏结束，顺利通过。

当棋盘所有数都填满，此时需要再进行添加滑块操作时，游戏结束，玩家失败。（一步操作包含“移动”、“合并”和“添加滑块”三个部分，在本步操作的“添加滑块”部分，判定结束失败条件）

当玩家键入 R 操作，并在[Y/N]中选择了 Y 重新开始时，游戏结束，恢复初始状态。

当玩家键入 Q 操作，并在[Y/N]中选择了 Y 退出游戏时，游戏结束，并关闭终端。

## 4 期末项目进阶

### 4.1 棋盘变化

在期末的项目中, 需要提示由玩家输入  $N$  和  $M$ , 表示一个  $N \times M$  的棋盘平面。

如果玩家没有输入, 则默认  $N=M=4$ 。

### 4.2 文件操作

#### 4.2.1 自定义输入序列

棋盘的填充内容, 传统的 2048 游戏由以下几种数字构成:  $\{2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048\}$ , 是一个长度为 11 的数字序列。在期末项目的游戏设计中, 可以由用户自定义多个文件 (大于 1), 由用户自定义序列的长度和序列的内容, 包括但不限于数字、英文、中文。

每个输入文件两行, 第一行表示输入序列的长度  $l$ , 第二行  $l$  个字符串, 从合并规则的从小到大, 表示输入的序列, 字符串之间以空格分割。

#### 4.2.2 输出操作序列

期末的项目中, 玩家需要记录每一步 WSAD 操作, 在游戏成功时, 将操作序列输出到文件中。

该输出文件包含两行, 第一行表示操作数  $t$ , 第二行  $t$  个字符 (WSAD 其中的一个), 表示每一步操作。

## 4.3 统计分数

### 4.3.1 基础分数

游戏成功时，基础分数为：默认分数 - 操作数分数。游戏失败时，基础分数为 0。

默认分数计算表达式为  $N * M * 10 * e^l$ ，其中 N 和 M 为棋盘的大小，e 为自然对数，l 为自定义输入序列的长度。

操作分数计算表达式为  $\log_2^t$ ，其中 t 为操作的步数。

### 4.3.2 连击奖励

对于连续的多次操作，如果每次操作都有一次成功的合并操作，则连击次数加一。如果该次操作没有合并操作，则连击次数清零。

连击奖励的计算表达式为斐波那契数列(1 1 2 3 5 8 13 …)。

### 4.3.3 多合奖励

如果一次操作，连续合并了多组数据，将会有额外的多合奖励。具体的计算方式如下：

假设一次操作，合并了 x 组数据，其中第 i 组合并操作 ( $1 \leq i \leq x$ )，合并的是输入序列的第 j 个数字 ( $1 \leq j \leq l$ )，设  $a(i)=j$ 。该次多合操作的奖励为：

$$\sum_{i=1}^x a(i) * e^{a(i)}$$

举例而言，假设一个 3\*4 的棋盘，输入序列的前四个数字分别是 2、4、8、16。

0 2 4 4 → 0 0 2 8

2 2 4 4 → 0 0 4 8

8 8 8 16 → 0 8 16 16

该次“D”操作后，合并了  $x=4$  组数据，第一组合并操作合并了第  $j=2$  个数字，第二组合并操作合并了  $j=1$  个数字，第三组合并操作合并了  $j=2$  个数字，第四组合并操作合并了第  $j=3$  个数字。该次操作的多合奖励得分为：

$$2 * e^2 + 1 * e^1 + 2 * e^2 + 3 * e^3$$

## 4.3.4 分数显示

### 4.3.4.1 屏幕显示

对于基础分数，当游戏结束时，终端显示“基础分数”。

对于连击奖励，每次操作的时候，终端显示“已连击次数”和“当前连击奖励分数”。当游戏结束时，终端显示“连击奖励分数总和”。

对于多合奖励，每次操作的时候，终端显示“合并次数”和“单次多合奖励分数”。当游戏结束时，终端显示“多合奖励分数总和”。

当游戏结束时，终端显示“游戏总得分”，为“基础分数” + “连击奖励分数总和” + “多合奖励分数总和”。

#### 4.3.4.2 文件输出

输出分数文件为 “score.txt” 。

文件前四行分别为 “基础分数”、连击奖励分数总和”、“多合奖励分数总和”和 “游戏总得分”。

文件第五行和第六行，记录 “最高连击数” 和 “最高连击得分”。

文件第七行和第八行，记录 “最高多合次数” 和 “最高多合得分”。

### 4.4 自动运行

当玩家键入 G 操作，并在随后的提示中输入一个数字 x 后，棋盘随机运行 x 步 WSAD 操作。此 x 步 WSAD 操作，效果等同于手动输入。

## 5 项目评分

本项目满分 100 分，包括两个部分：基础功能部分、综合评价部分。其中，基础部分评分共 70 分，综合部分评分共 30 分。

### 5.1 基础部分评分

说明	分值
成功输入 N*M 的棋盘	5



自定义输入序列	10
输出操作序列	5
统计分数正确	15
游戏结束、重启、运行正确	30
自动运行	5

## 5.2 综合部分评分

说明	分值
设计文档（包括但不限于程序结构的设计和分析，主要函数功能，如何使用你的程序，不同运行结果的展示，总结和感悟）	10
代码规范（包括但不限于变量的使用，缩进与换行，备注）	5
程序鲁棒性（对于错误的输入，能否有错误判断，并提示错误信息或者忽略错误输入）	5
面试展示（能否清晰讲解项目逻辑，能否回答助教提问）	10

## 6 提交

### 6.1 提交物

请将你的源代码和设计文档打包好，命名为学号\_姓名（如 20210240004\_郭天成），作为提交物，提交到超星学习通对应的作业中。

### 6.2 截止时间

第 17 周结束。第 18 周面试。

## 7 注意事项

1. 认真做好每个功能点，尤其是基础部分的功能。
2. 合理安排时间，尽早开始代码实践，不要拖到最后。
3. 欢迎同学们相互讨论，但是抄袭是严格禁止的。一旦发现抄袭行为，抄袭者和被抄袭者都根据抄袭量酌情扣分（甚至直接给予 0 分）。