**C语言项目报告**

**项目名称：\_\_\_\_B2B（Ball to Basket）\_\_\_**

**项目成员：\_\_\_\_\_申奥、朱子琦、高庆霖\_\_\_\_\_\_**

**填写日期：\_\_\_\_\_\_2021.07.09\_\_\_\_\_**

# **摘要（Abstract）**

B2B（Ball to Basket）游戏项目是一款应用Unity的物理引擎的多种功能的休闲益智类游戏，指能在游戏中锻炼人的逻辑力与敏捷力的一系列需利用部分物理知识才能实现胜利的游戏。

Unity作为一个强大的游戏开发引擎，具有较为完善的有关物体碰撞等一系列游戏所需要的函数，以及较为便捷的UI设计方法。故我们选择使用unity开发一个有关小球运动与碰撞的物理休闲益智类游戏，并在其中加入诸如混色、叠加等设计增加其可玩性，再配以设计的烧脑关卡达到寓教于乐、在游戏中锻炼逻辑的目的，取名为B2B（Ball to Basket）。

# **问题描述（Problem Statement）**

* 1. **项目背景**

近年来，随着经济科技的发展、生活节奏的加快，人们对休闲类游戏的需求越来越大。物理休闲益智类游戏使人们在休闲放松的同时能够锻炼自己的逻辑力与敏捷力，是一项很好的放松选择。现在市面上有许多物理休闲益智类小游戏，如割绳子、弹珠等，相关游戏设计思路与设计方法均有较强的借鉴意义。本游戏就是在这种背景下被开发出来的，我们致力于实现游戏放松身心目的的同时，能对玩家的大脑及能力实现一定的锻炼与提高。

对于游戏设计环境的选择，我们参考了班内大多数同学的使用Unity的现状，基于多交流多学习共同进步的目的和Unity作为一个强大的游戏开发引擎，具有较为完善的有关物体碰撞等一系列游戏所需要的函数，以及较为便捷的UI设计方法的重要优势，我们选择了Unity作为游戏制作引擎。

* 1. **实际问题**

但在实际工作中我们遇到了游戏物体与交互道具绘制来源、**vs函数的关联、多颜色小球的实现与交互、物理引擎的使用、胜利失败条件的判定、游戏体验的丰富、益智烧脑关卡设计、开局界面的设计等诸多细节问题。**通过灵活使用图片处理类软件和编程软件，以及利用的游戏引擎中的部分函数，并查找资料，参考学习源代码，灵活使用算法，我们成功解决了上述问题，并且通过开关、加速器、翻斗、篮子堆叠、染色器、混色器等多种游戏道具实现丰富的游戏关卡，使项目可玩性和完整度大大提高。

# **组内分工（Group Division）**

|  |  |  |  |
| --- | --- | --- | --- |
| 名字 | 学号 | 学院 | 工作百分比 |
| 申奥 | 09201231 | 生物 | 35% |
| 朱子琦 | 13200303 | 生物 | 30% |
| 高庆霖 | 22299221 | 生物 | 35% |

* 1. **申奥**

申奥是小组的组长，进行了项目的选题，安排了项目的分工，建立了项目的框架。申奥主要负责这个程序中的物理碰撞的实现，开关、加速器和翻斗三个道具的功能实现和预制，以及第3、5、6、7、8、9关卡的制作。

* 1. **朱子琦**

朱子琦主要负责UI布局与通用预制的生成，篮子堆叠、染色器和混色器三个道具的功能实现和预制，以及第2、4、10、11、12关卡的制作。

* 1. **高庆霖**

高庆霖辅助进行了项目的选题和游戏制作引擎的选择。高庆霖主要负责了游戏胜利失败的判定，界面功能按钮的实现和开局界面与选关界面的实现。最后，他整合了其他小组成员编写的项目并生成了游戏应用程序。

* 1. **文档工作**

朱子琦负责了开题、中期和期末答辩PPT的制作和开题项目开发设计书；

高庆霖负责了中期的接口设计文档和软件需求文档；

申奥负责了中期的详细设计文档和期末的项目报告。

# **分析（Analysis）**

* 1. **游戏逻辑与游戏规则**

首先，我们需要确定游戏逻辑与游戏规则。游戏的目标十分简单，即玩家通过在合适的入口按照合适的顺序放置带有颜色的小球，使小球在物理引擎的模拟下能够掉落入对应颜色的篮子里，即“Ball to Basket”。

概括游戏规则如下:

(1)拖动颜色小球按钮到入口区生成物理小球

(2)玩家需要使带颜色的小球进入对应篮子中赢得胜利

(3)篮子中有多于一个小球或落入小球颜色错误则游戏失败

(4)在可能短的时间内完成关卡从而得到更高的星级和金钱奖励

* 1. **界面建立与游戏按钮与路径绘制**

建立游戏界面、开局界面、选关界面、胜利界面、两个失败界面，绘制各种游戏按钮与路径。

* 1. **通用功能**

对每一关均需要成立的功能需求如下：暂停功能、重玩功能、小球生成功能、小球拖拽功能、小球销毁功能、生成物理小球功能、小球轨道设置功能、篮子承接功能、胜利失败判定功能、计时器与积分功能。

* 1. **特有道具功能**

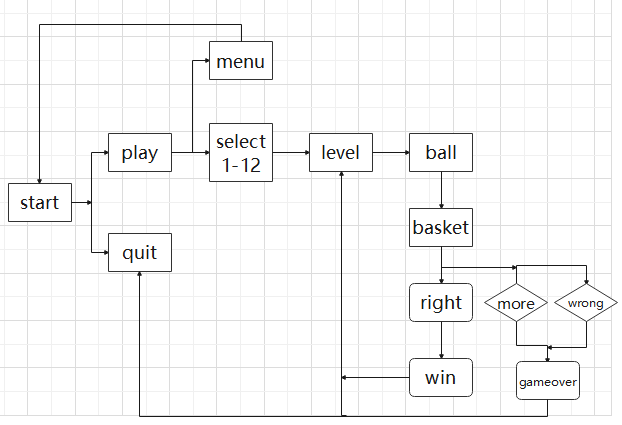
对特定关卡新引入的道具的功能需求如下：开关功能、篮子堆叠功能、翻斗功能、染色器与混色器功能、加速器功能。

* 1. **游戏整体功能**

对关卡游戏的连续性与完整性的功能需求如下：退出游戏功能、切换界面功能。

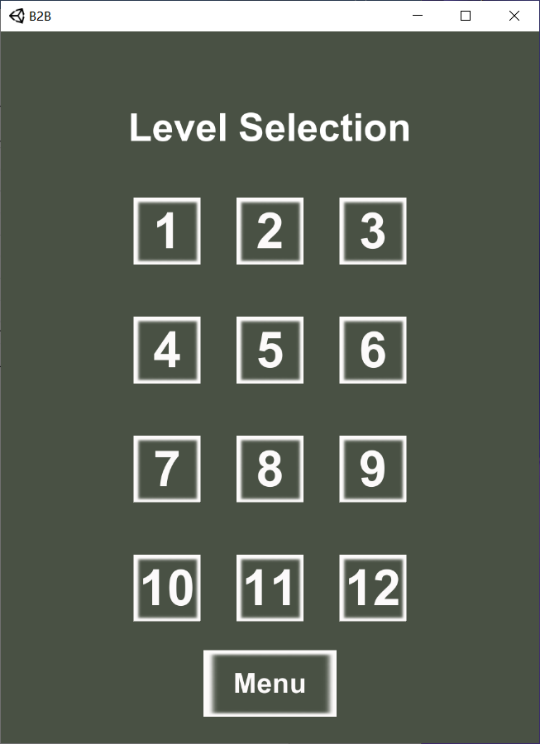
# **设计（Design）**

* 1. **游戏逻辑与游戏规则**

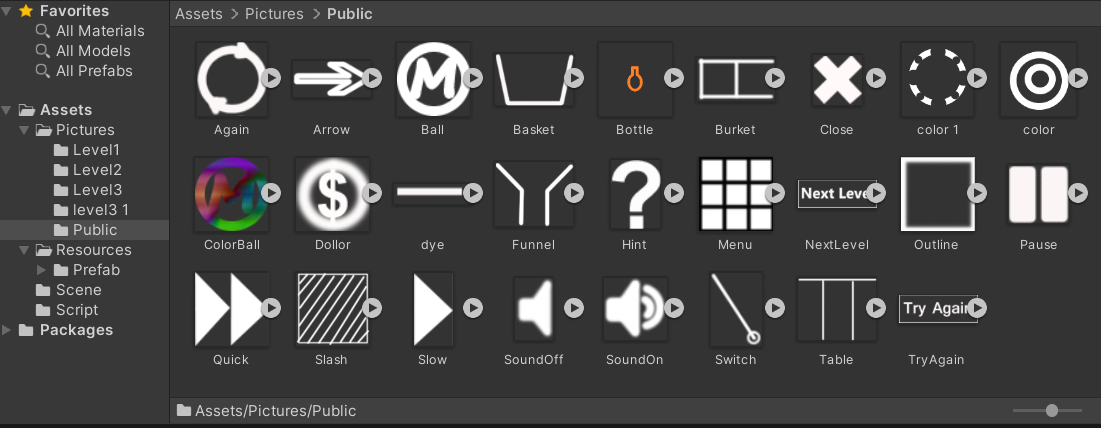


**图一**

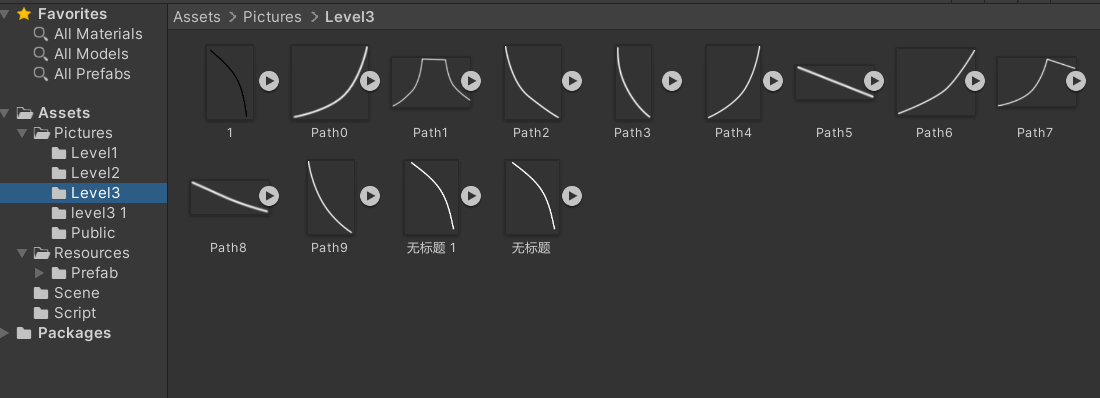
**界面建立与游戏按钮与路径绘制**



**图二 开局界面 图三 选关界面**



**图四 游戏道具**



**图五 游戏路径**

* 1. **通用功能**

暂停功能：由暂停按钮实现游戏中计时器的暂停与开始

重玩功能：由重玩按钮实现游戏中当前关卡的重玩

小球生成功能：由小球按钮实现生成对应颜色的小球

小球拖拽功能：由鼠标位置检测函数实现生成小球随鼠标移动的拖拽效果

小球销毁功能：由鼠标事件检测函数实现鼠标左键松开时生成小球的销毁

生成物理小球功能：由3D/2D射线碰撞判定实现在入口区生成物理小球

小球轨道设置功能：由刚体组件属性和物理碰撞检测实现小球沿设计轨道运动的物理碰撞过程

篮子承接功能：边碰撞器或多边形碰撞器检测实现小球在篮子底部停止

胜利失败判定功能：由胜利失败情景枚举函数与物理属性检测器实现游戏关卡胜利或失败的判定

计时器与评分功能：由总体控制脚本中的二维数组判定每关通关时间规定星级标准，由胜利界面脚本中的内置函数计时

* 1. **特有道具功能**

开关功能：由位置差异的组合体与碰撞函数实现开关切换功能

篮子堆叠功能：由篮子组碰撞函数与篮子碰撞体实现篮子对不同小球经过时的穿过判定

翻斗功能：由刚体旋转函数与翻斗碰撞体实现翻斗对小球的承接与倒出

染色器与混色器功能：由颜色索引函数与转换混合函数实现对小球颜色的改变

加速器功能：由位置差异的组合体与加速函数实现对小球速度与方向的改变

* 1. **游戏整体功能**

退出游戏功能：由退出游戏按钮实现退出游戏

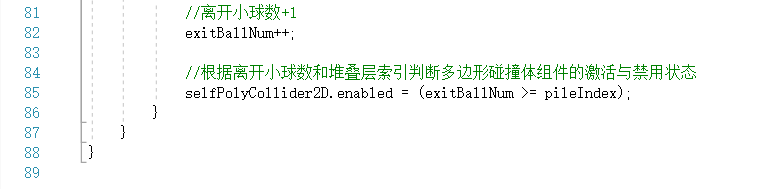
切换界面功能：由场景索引函数与场景按钮实现切换场景

# **实施（Implementation）**

* 1. **Basket脚本：实现篮子颜色设置功能、篮子堆叠功能、篮子承接小球功能、篮子状态函数赋值**

****

****

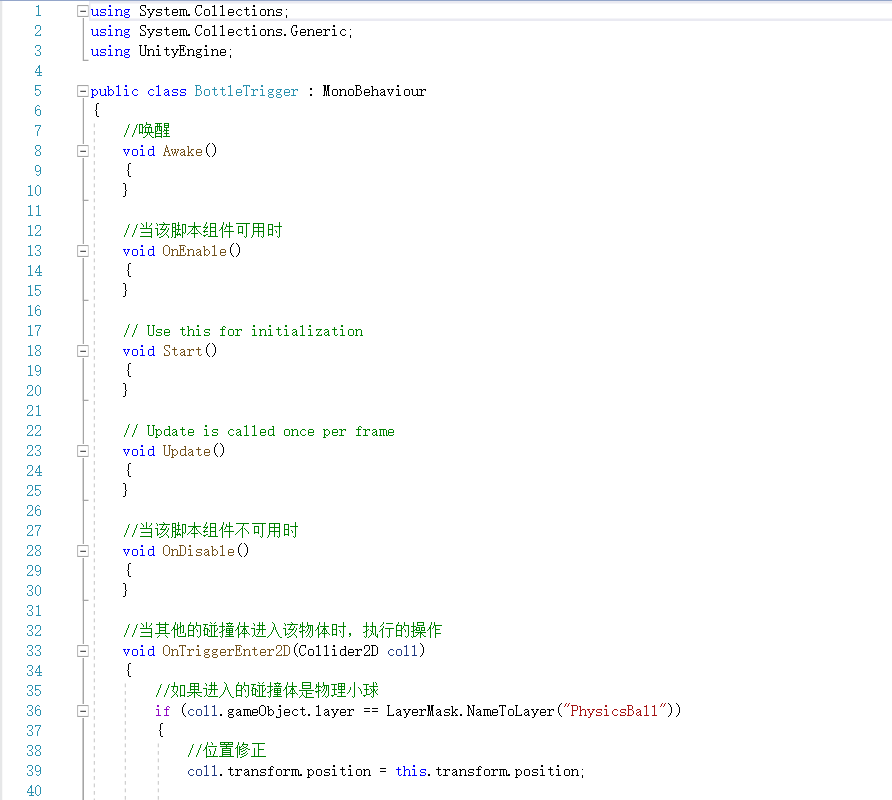
****

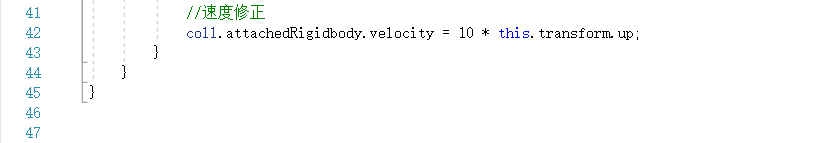
* 1. **Bottle脚本：实现瓶子状态切换功能**

****

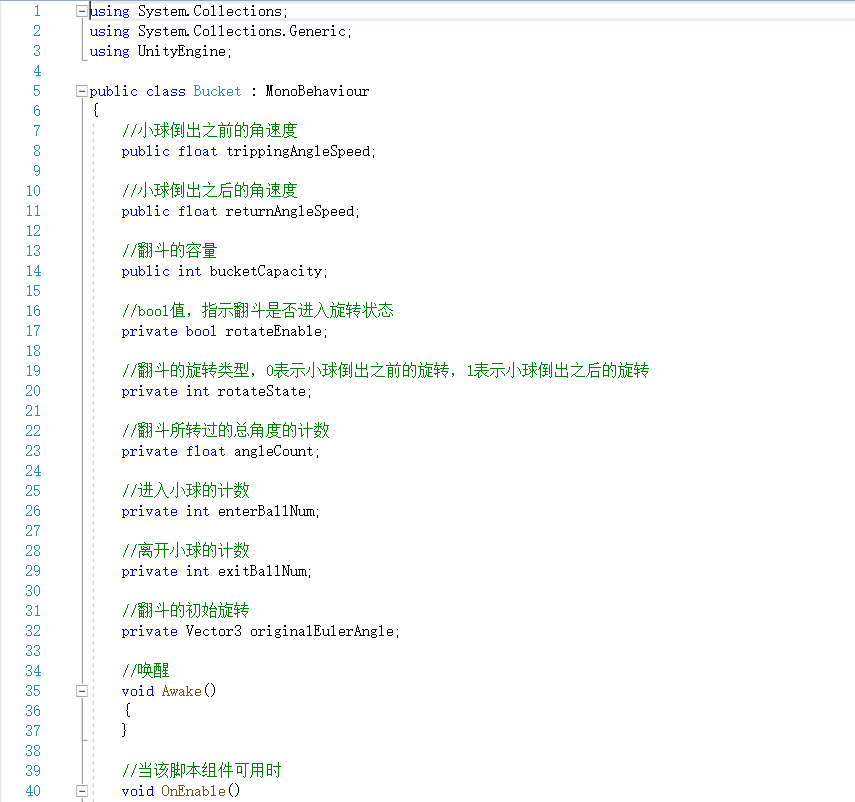
****

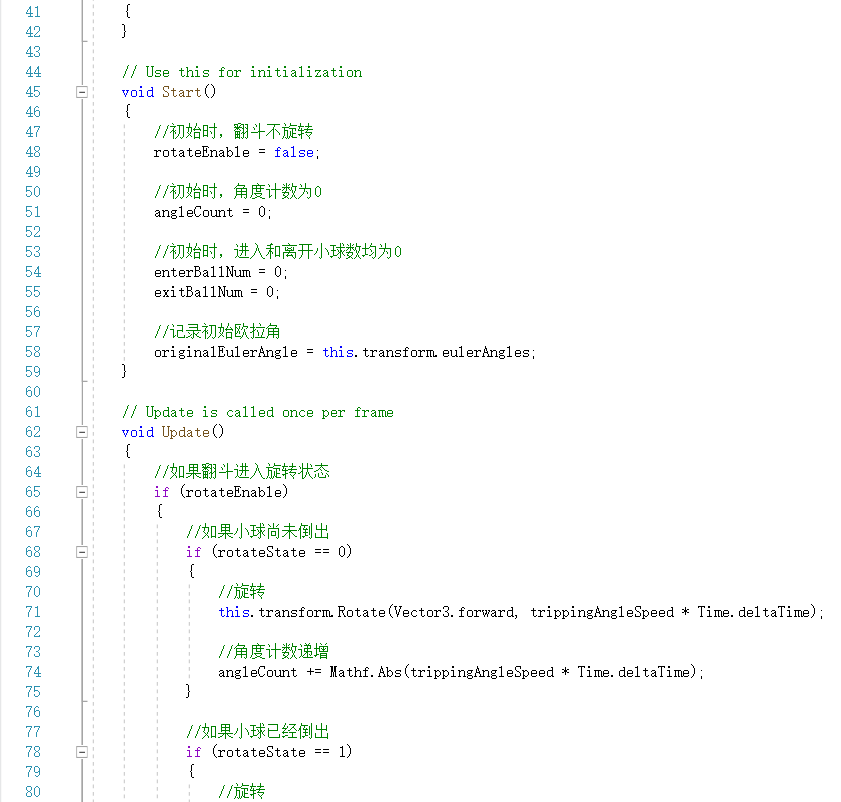
* 1. **Bottle Trigger脚本：实现瓶子对小球运动方向和速度的改变功能**

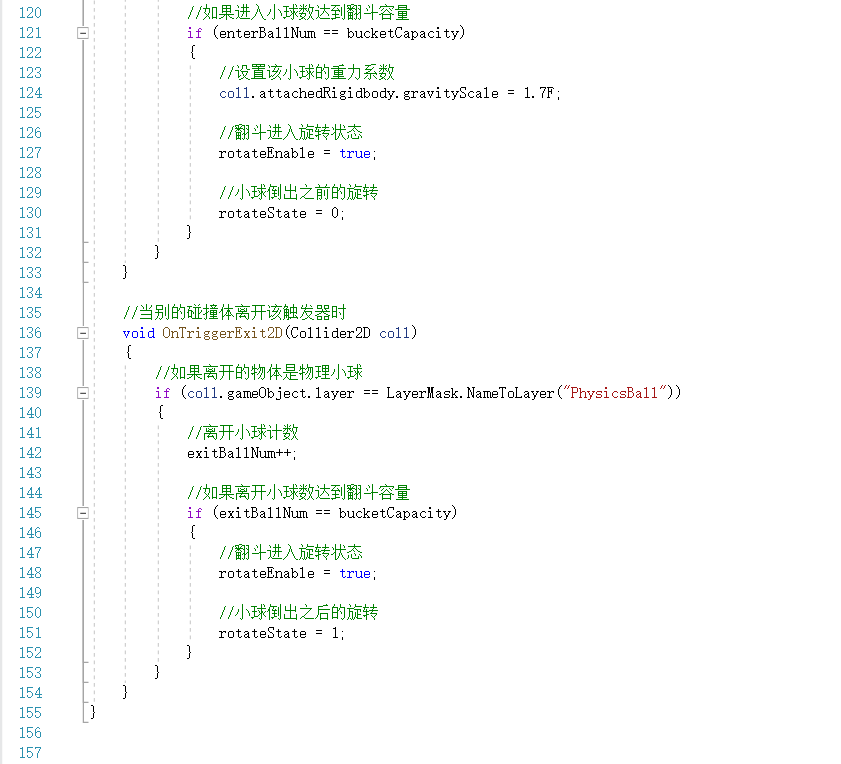
****

****

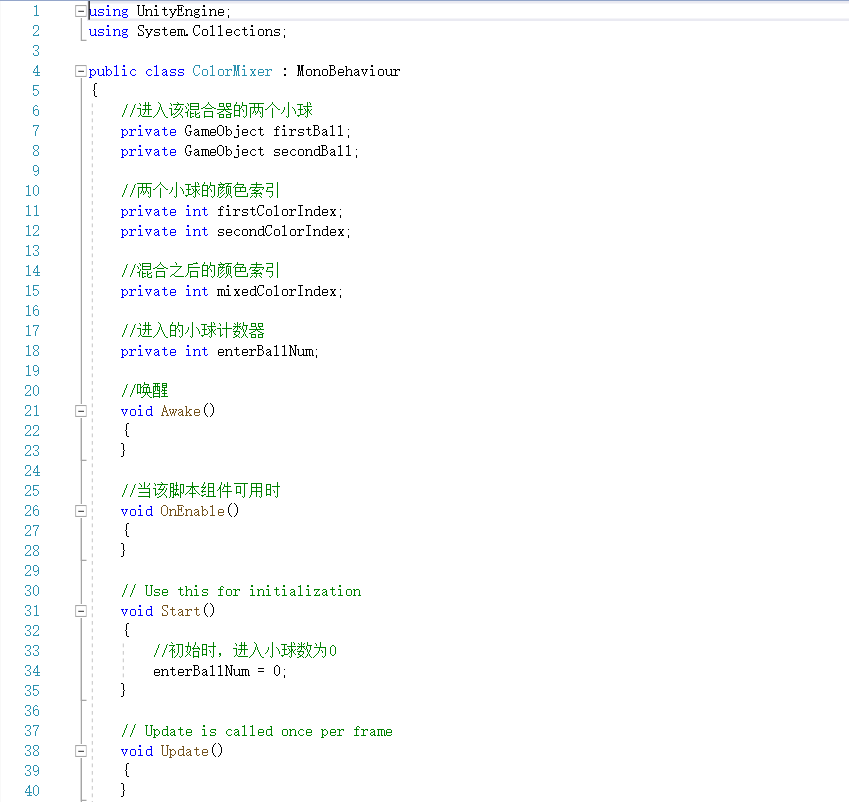
* 1. **Bucket脚本：实现翻斗状态的赋予功能、进入翻斗小球的计数功能、翻斗旋转方向与速度的设定功能、倒出小球重力系数修正功能**

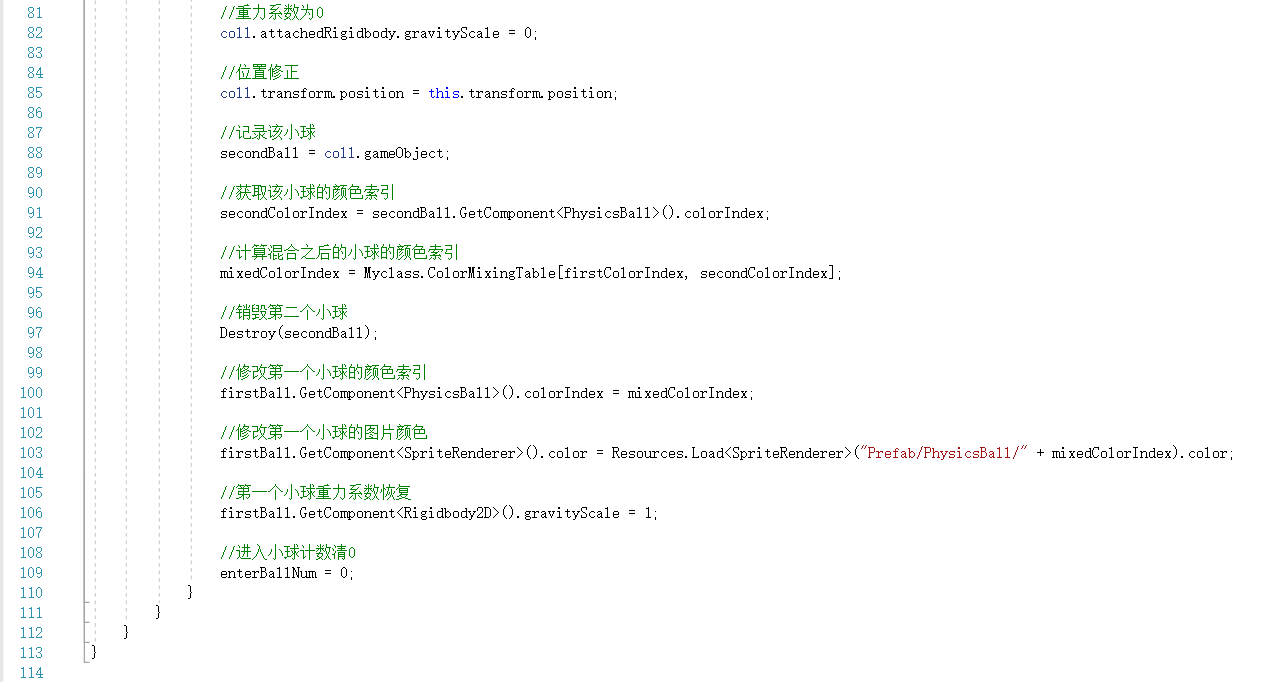
****

****

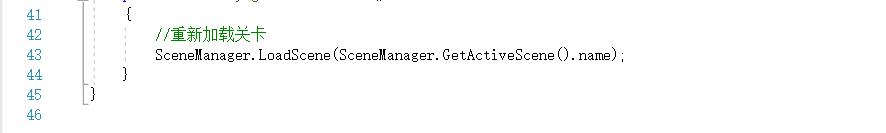
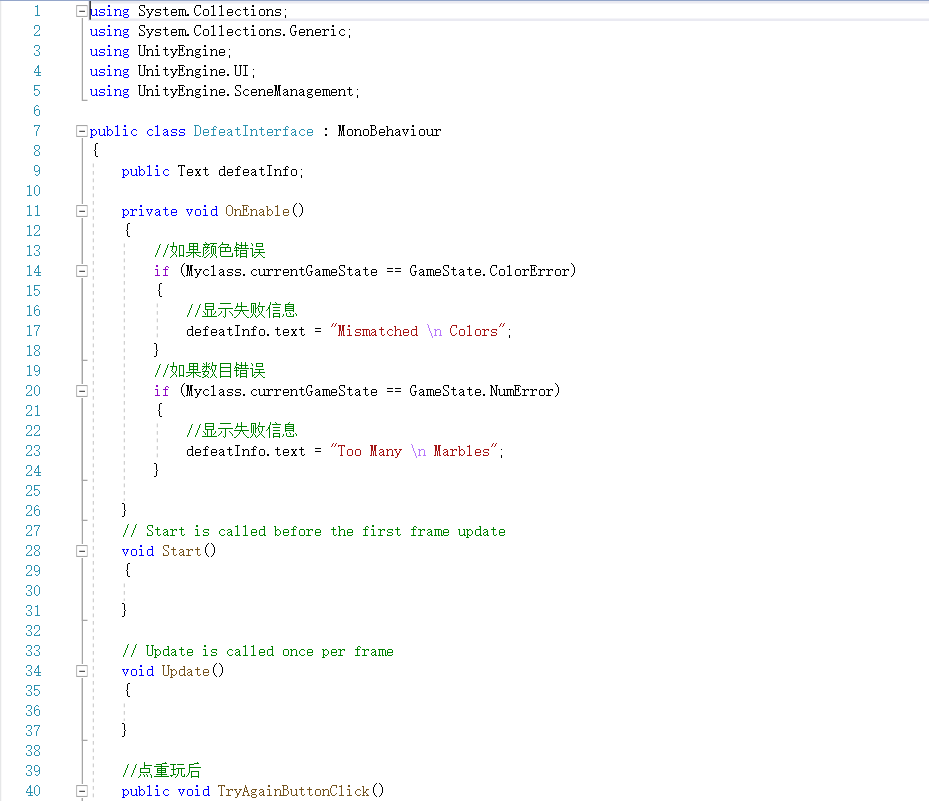
****

* 1. **Color Mixer脚本：实现获取进入混色器小球颜色索引功能、进入小球数量计数功能、混色后新小球颜色赋予功能、小球暂停与下落功能**

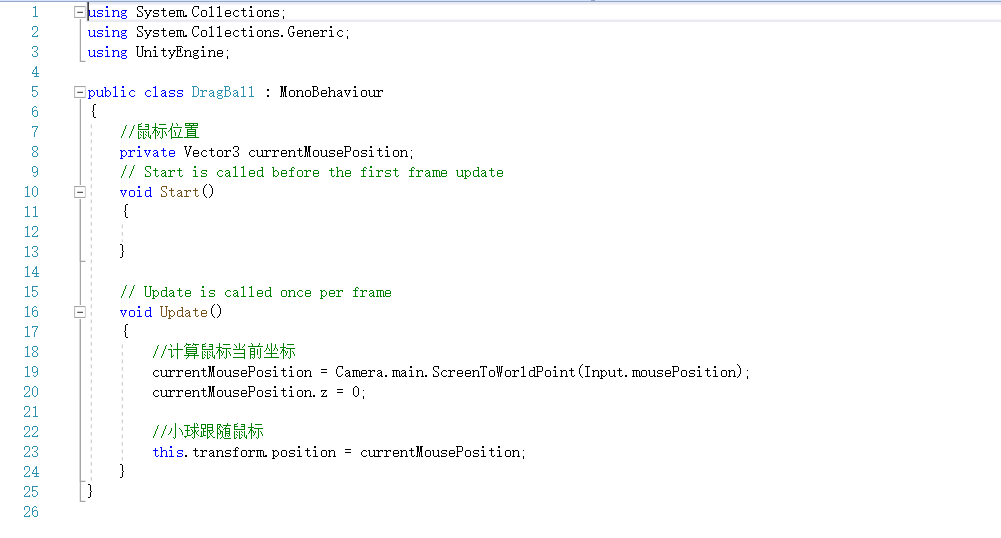
****

****

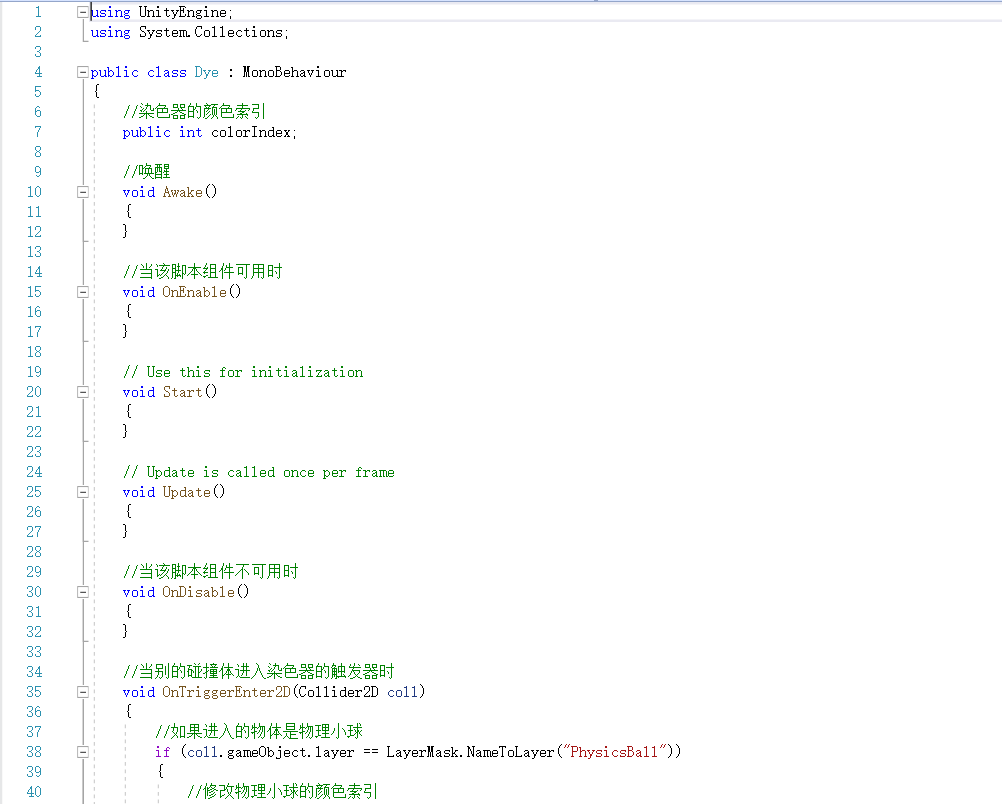
* 1. **Defeat Interface脚本：实现颜色错误信息显示功能、书面错误信息显示功能、失败后重玩按钮重玩功能**

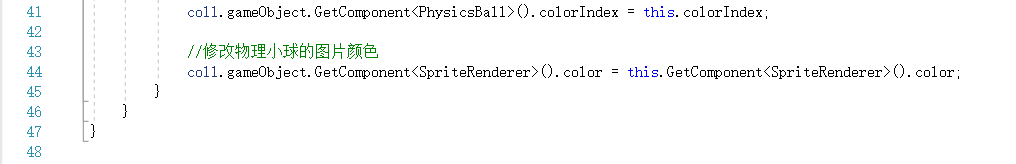
****

* 1. **Drag Ball脚本：实现鼠标点击拖动小球图标功能**

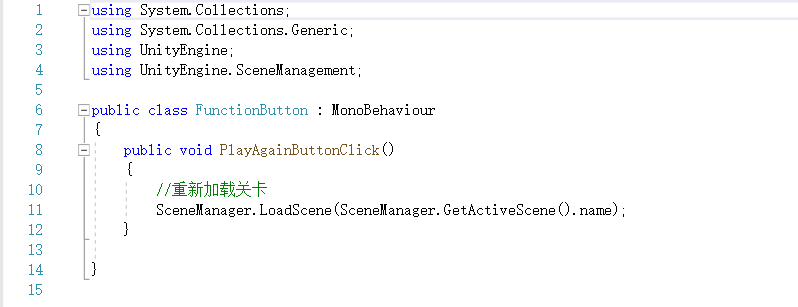
****

* 1. **Dye脚本：实现改变物理小球颜色功能**

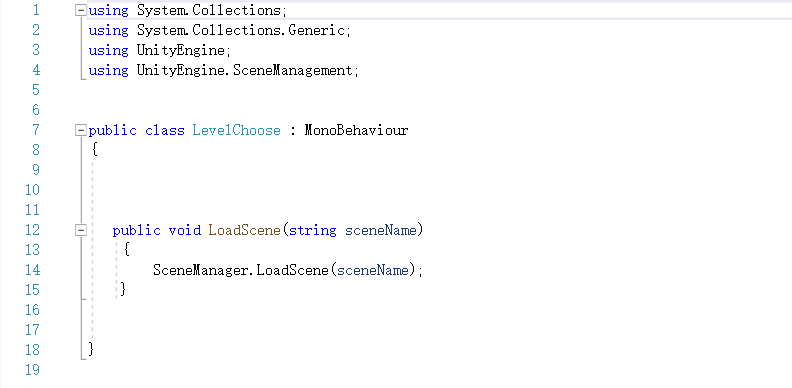
****

****

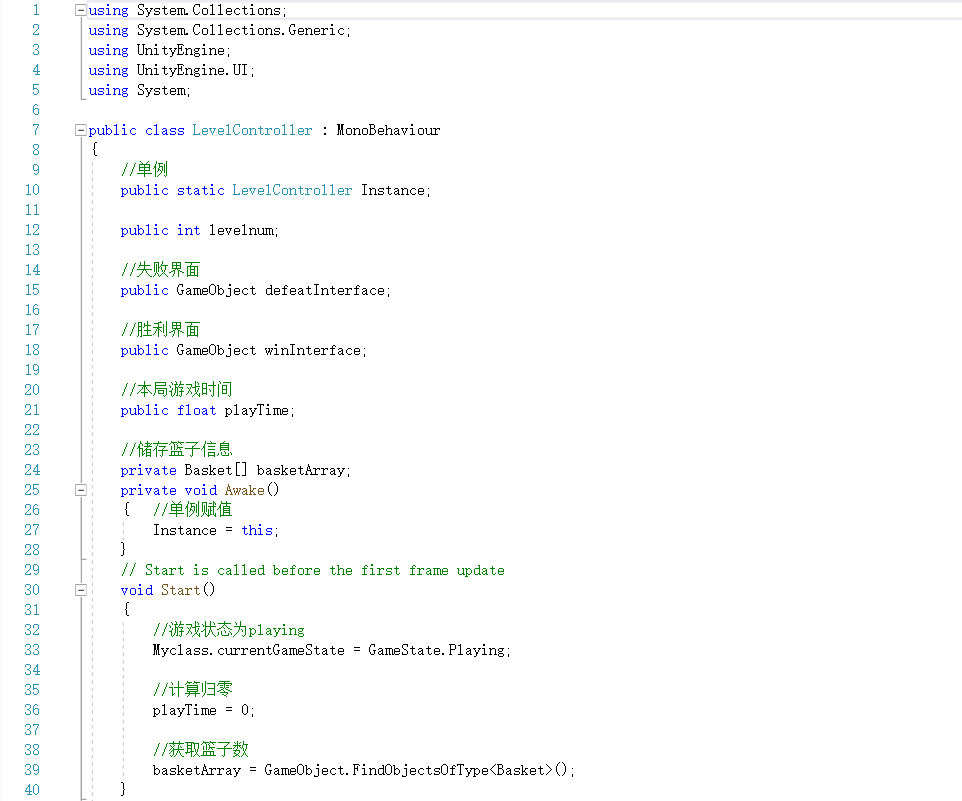
* 1. **Function Button脚本：实现游戏中重玩按钮重玩功能**

****

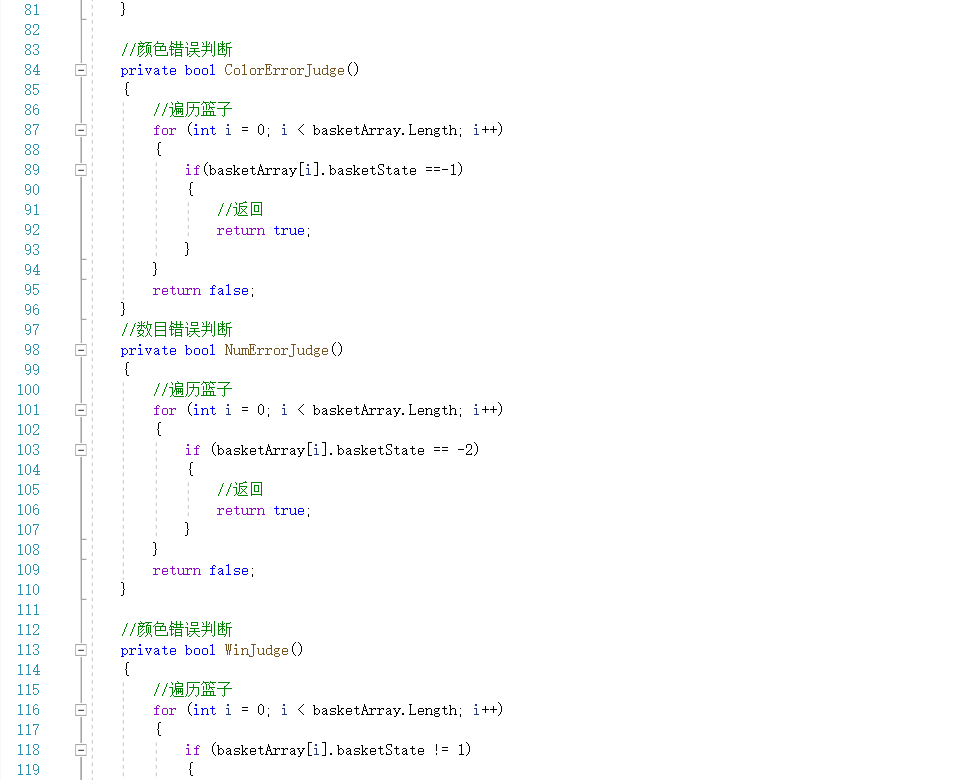
* 1. **Level Choose脚本：实现选关界面选关按钮选关功能**

****

* 1. **Level Controller脚本：实现胜利判定与界面显示、数目错误判定与界面显示、颜色错误判定与界面显示功能**

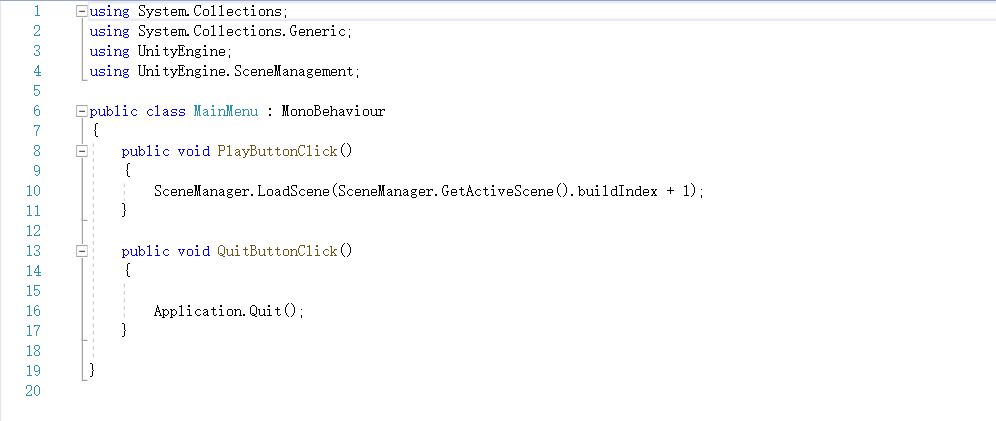
****

****

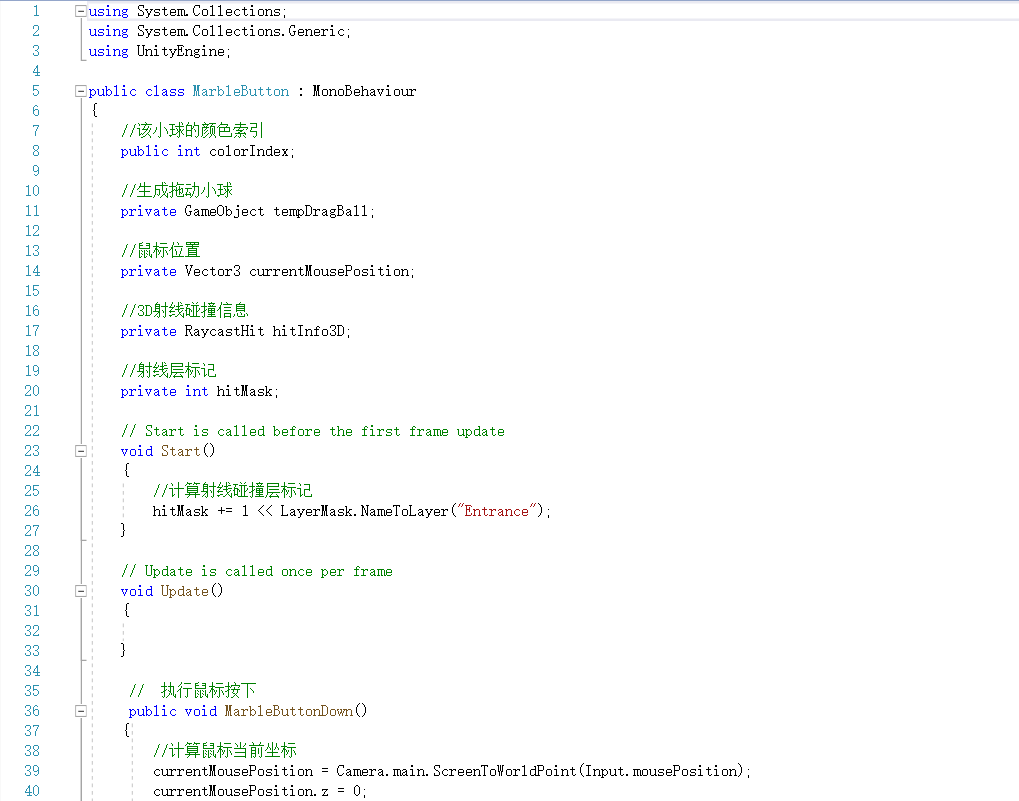
****

****

* 1. **Main Menu脚本：实现开始界面的进入与退出功能**

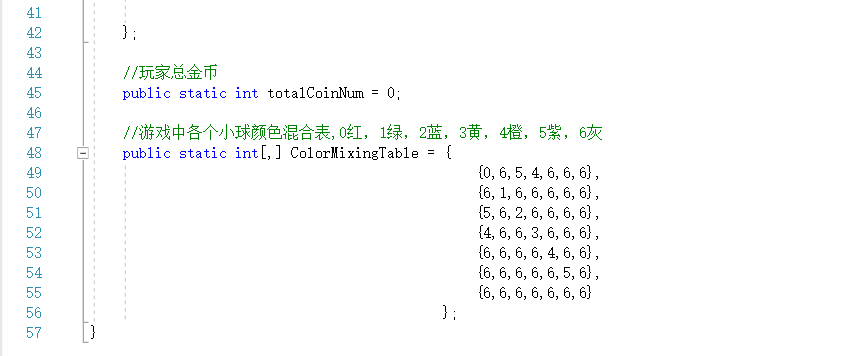
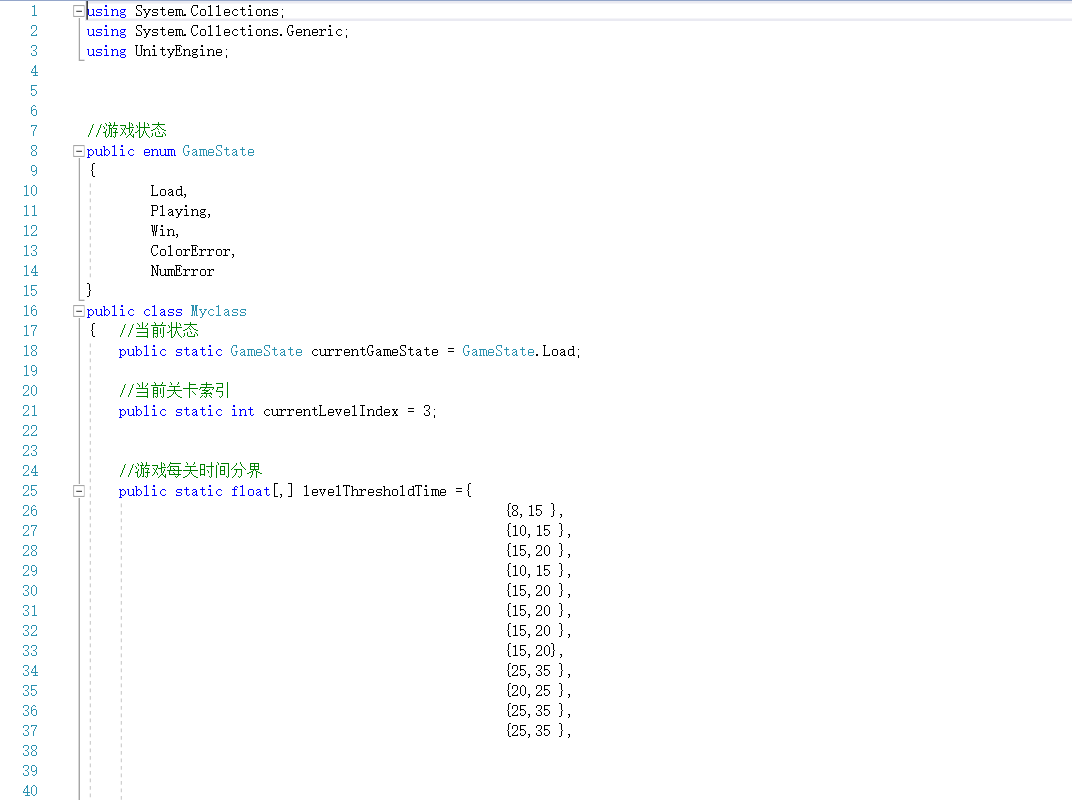
****

* 1. **Marble Button脚本：实现对应颜色拖动小球的生成与销毁、鼠标位置与状态的检测、对应颜色物理小球的生成功能**

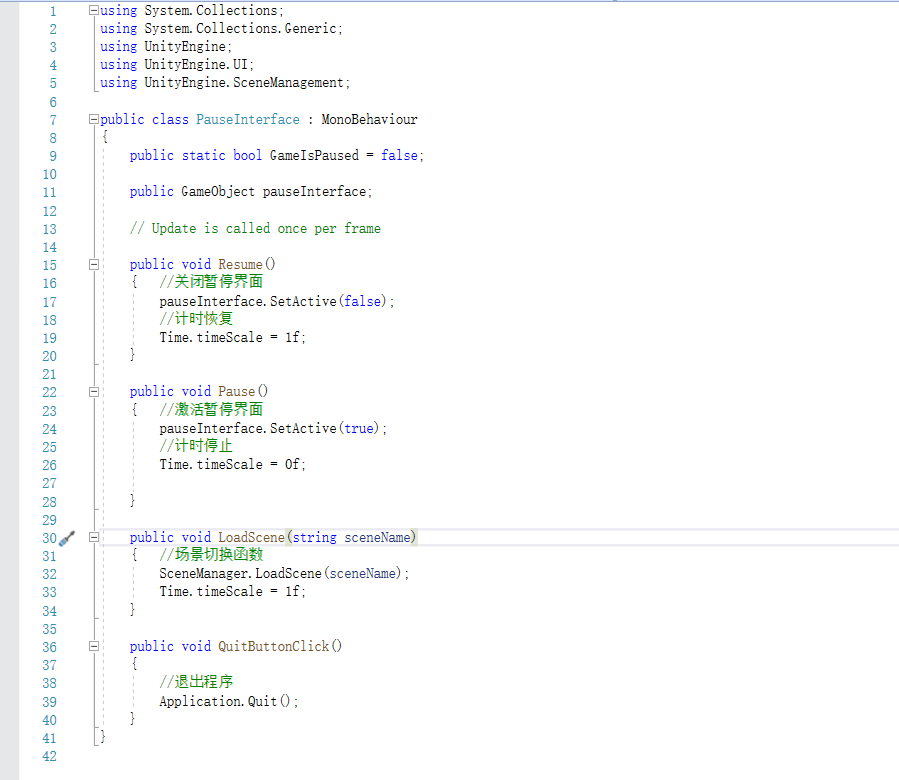
****

****

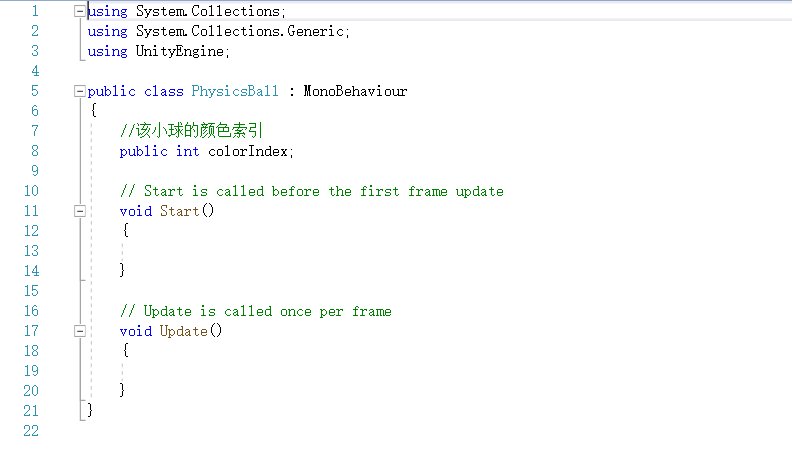
* 1. **My class脚本：实现游戏状态的赋予、关卡时间目标的设定、小球混色原理设定功能**

****

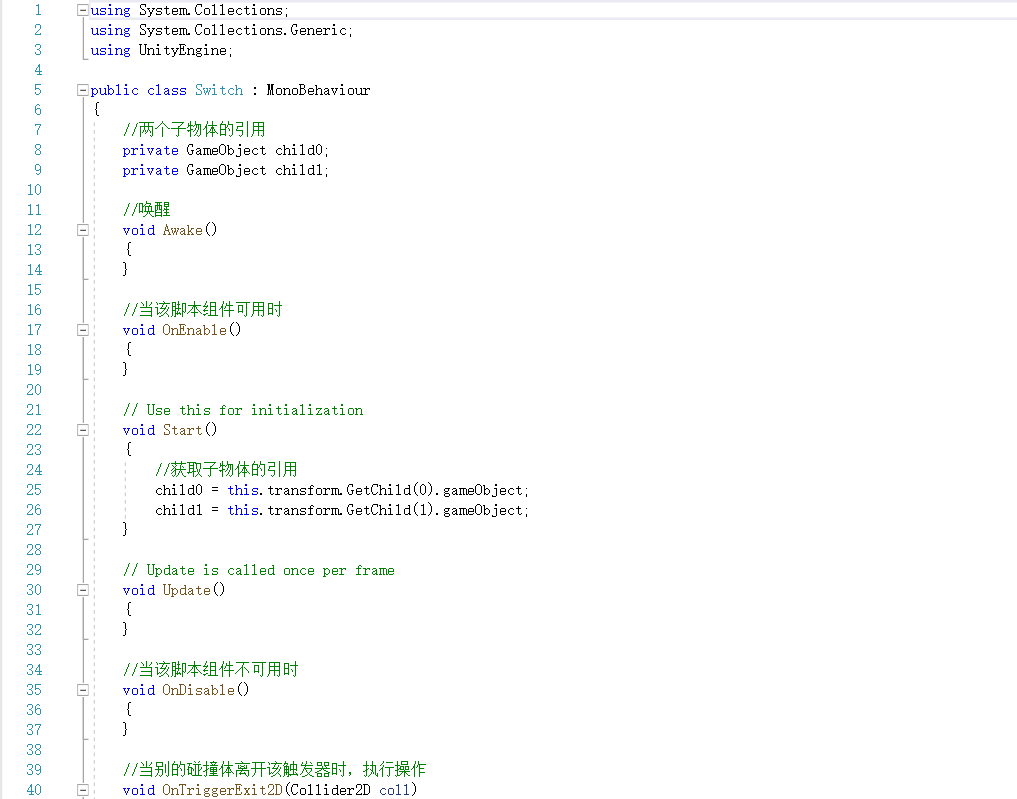
* 1. **Pause Interface脚本：实现暂停界面的激活与关闭、控制游戏时间的流动和停止功能**

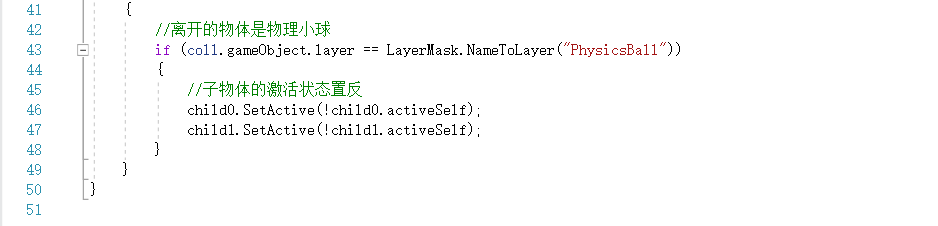
****

* 1. **Physics Ball脚本：实现物理小球状态的赋予功能**

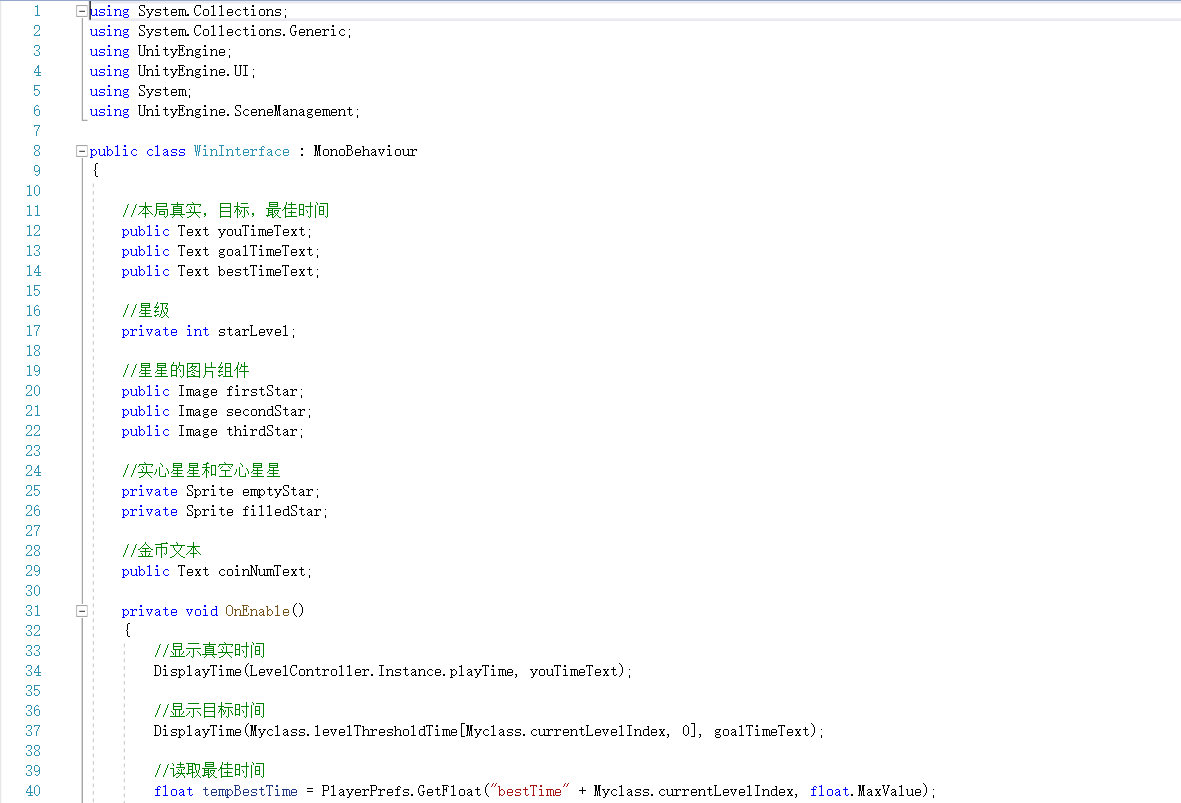
****

* 1. **Switch脚本：实现开关状态切换功能、物理小球路径检测功能**

****

****

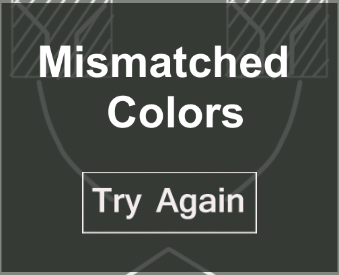
* 1. **Win Interface脚本：实现关卡计时、时间等级赋予、星星等级判定与显示、最佳时间的更新与显示、金币获取与累计、胜利界面关卡重玩功能**

****

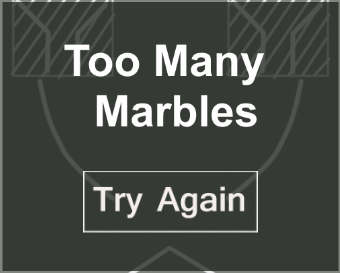
****

# **测试（Test）**

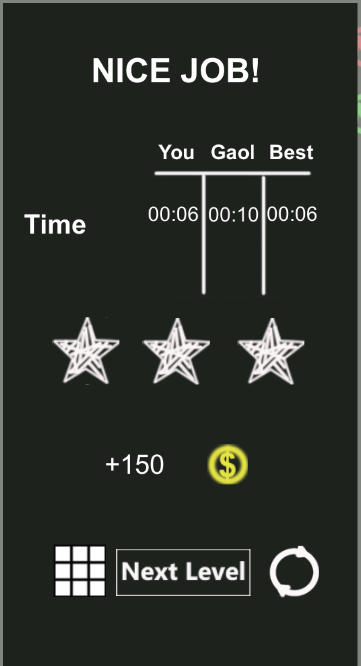
* 1. **功能测试**
     1. 颜色失败判定



* + 1. 数量失败判定

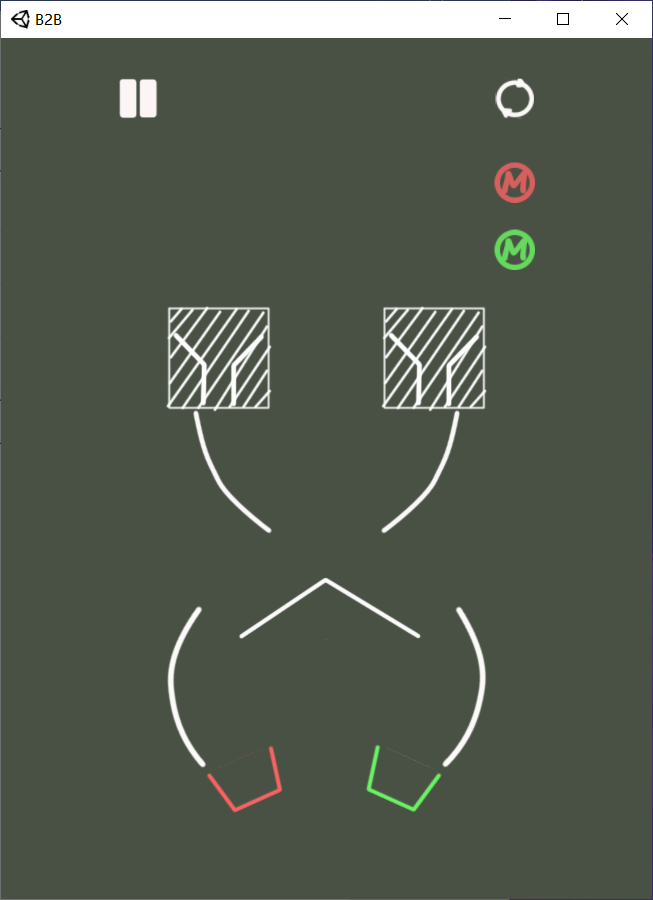


* + 1. 胜利判定



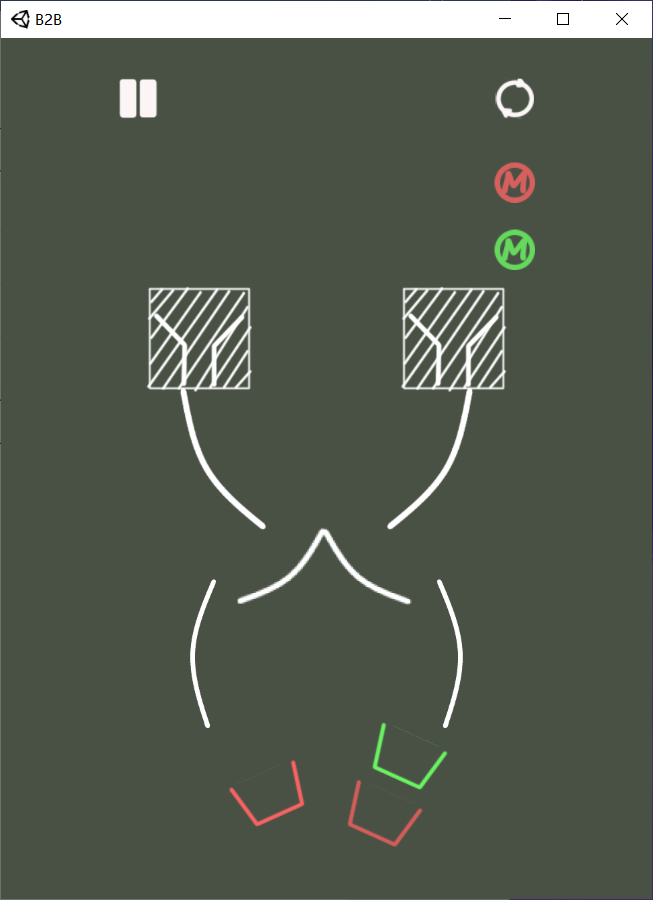
* 1. **关卡展示与攻略**

### Level 1



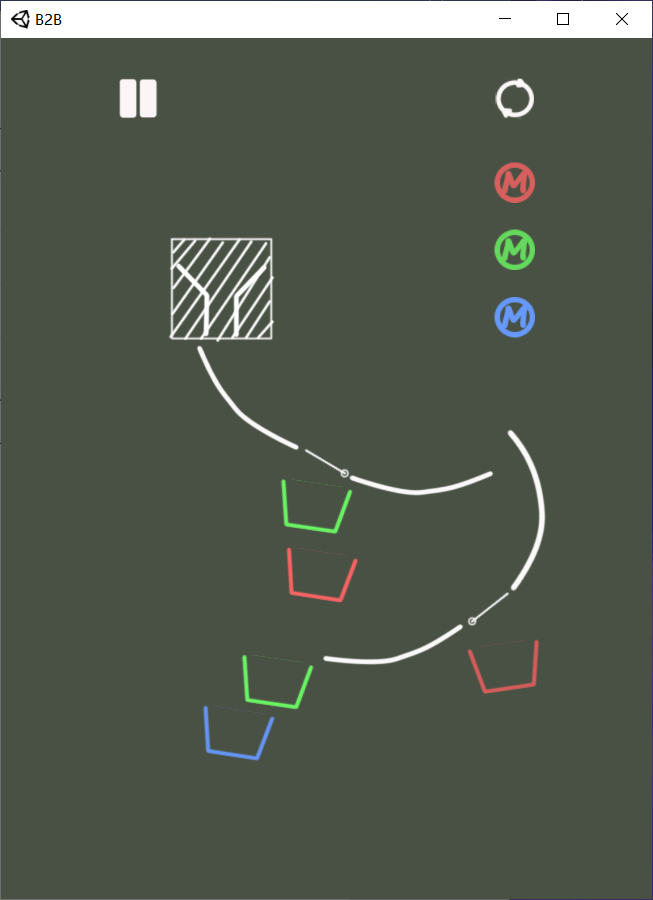
关卡攻略：左入口：绿→右入口：红。

* + 1. Level 2



关卡攻略：左入口：红→右入口：红→右入口：绿。

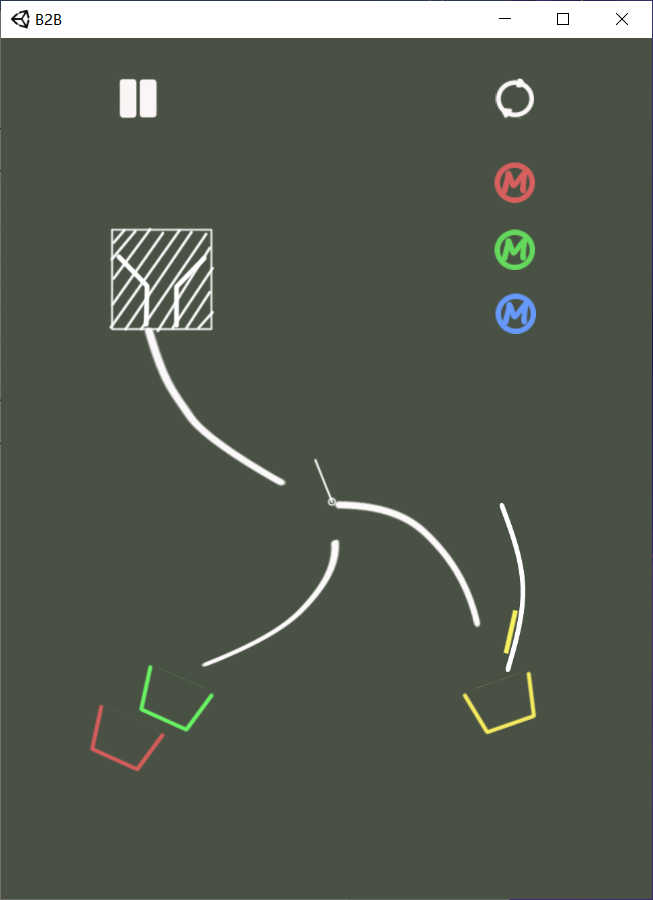
* + 1. Level 3



游戏攻略：入口：蓝→入口：红→入口：红→入口：绿

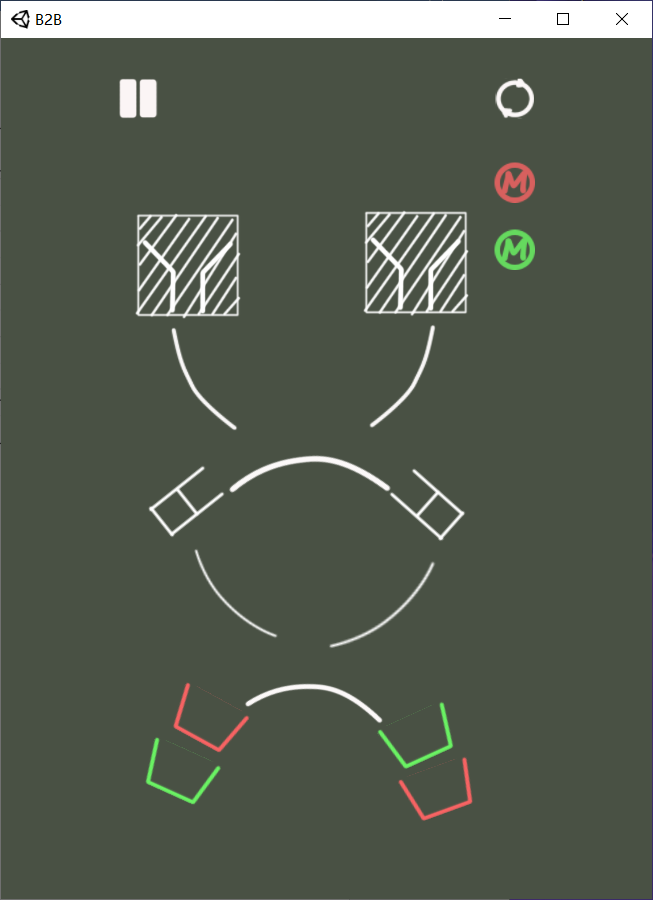
→入口：绿

* + 1. Level 4



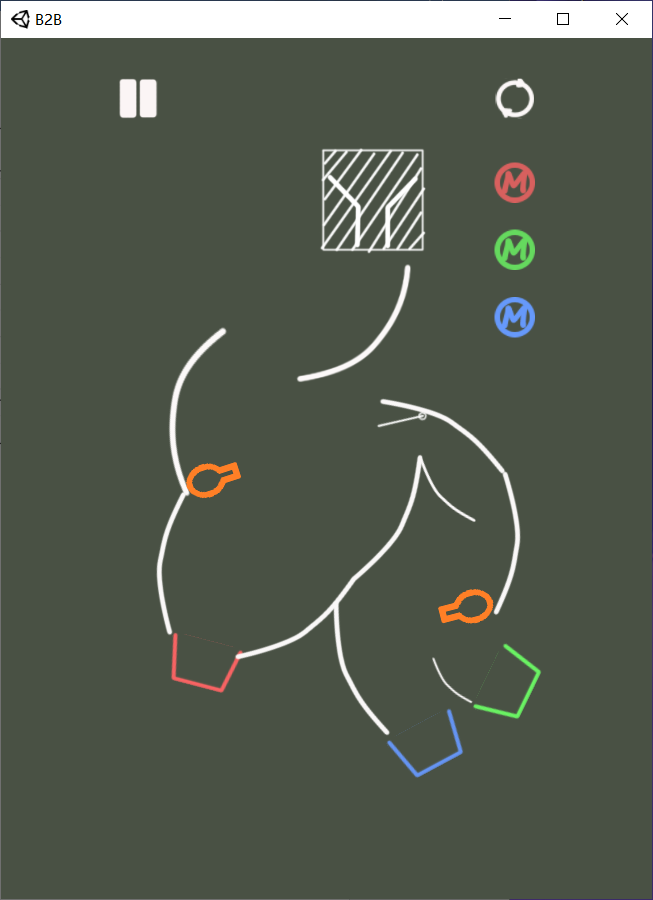
游戏攻略：入口：红→入口：黄→入口：绿

* + 1. Level 5



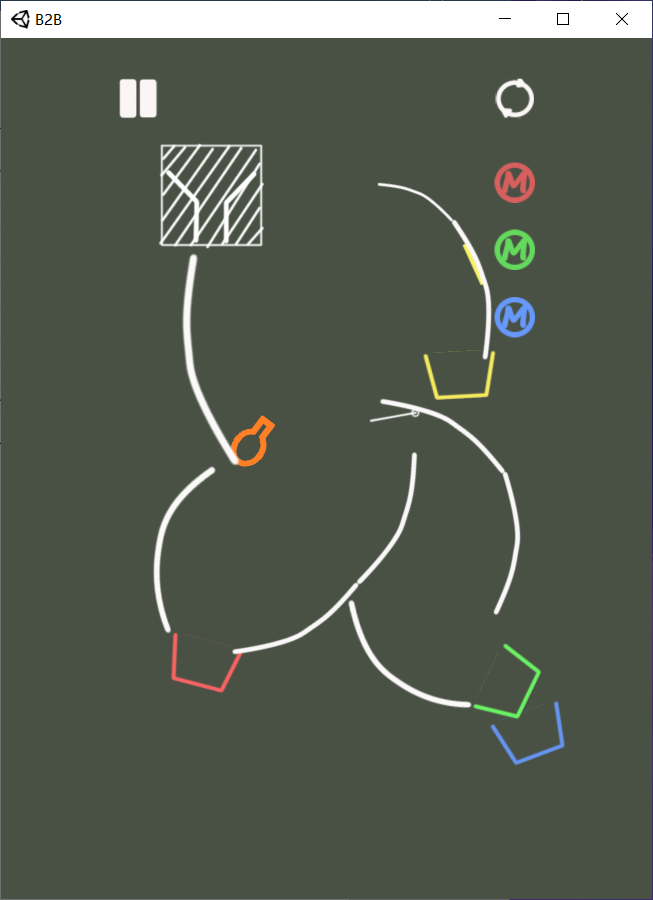
游戏攻略：左入口：红→左入口：绿→右入口：绿→右入口：红

* + 1. Level 6



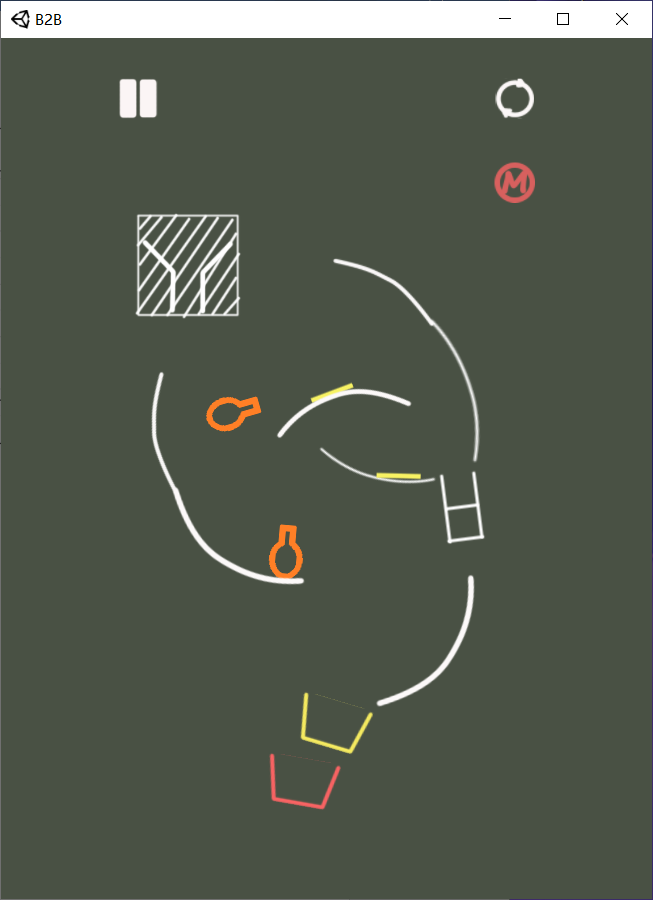
游戏攻略：入口：蓝（左加速瓶1右加速瓶1）→入口：红（左加速瓶1右加速瓶1）→入口：绿（左加速瓶1右加速瓶1）

* + 1. Level 7



游戏攻略：入口：任意（加速瓶1）→入口：蓝（加速瓶2）→入口：红（左加速瓶1右加速瓶1）

* + 1. Level 8



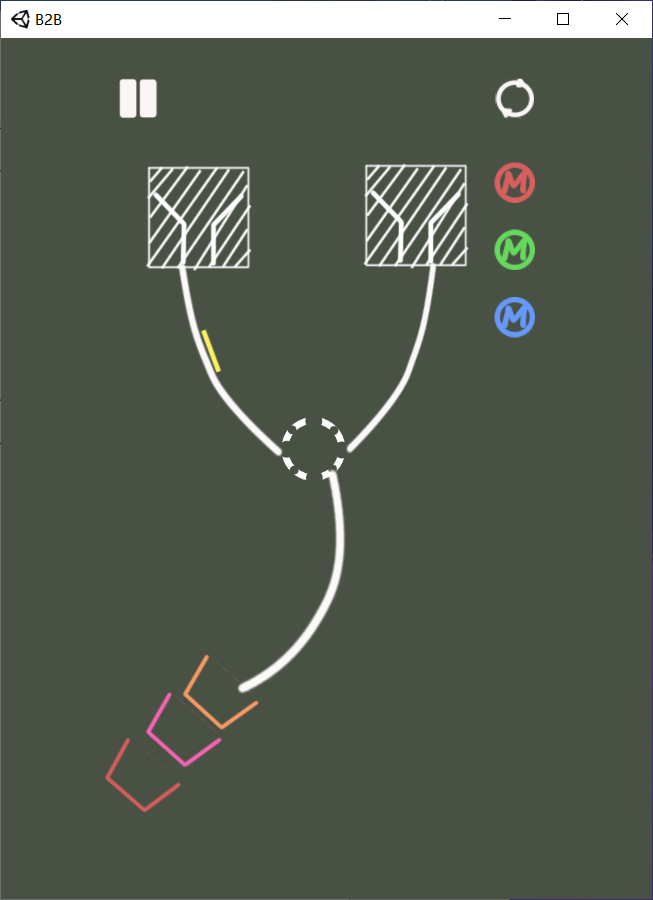
游戏攻略：入口：红（上加速瓶1）→入口：蓝（加速瓶2）→入口：红（左加速瓶1右加速瓶1）

* + 1. Level 9



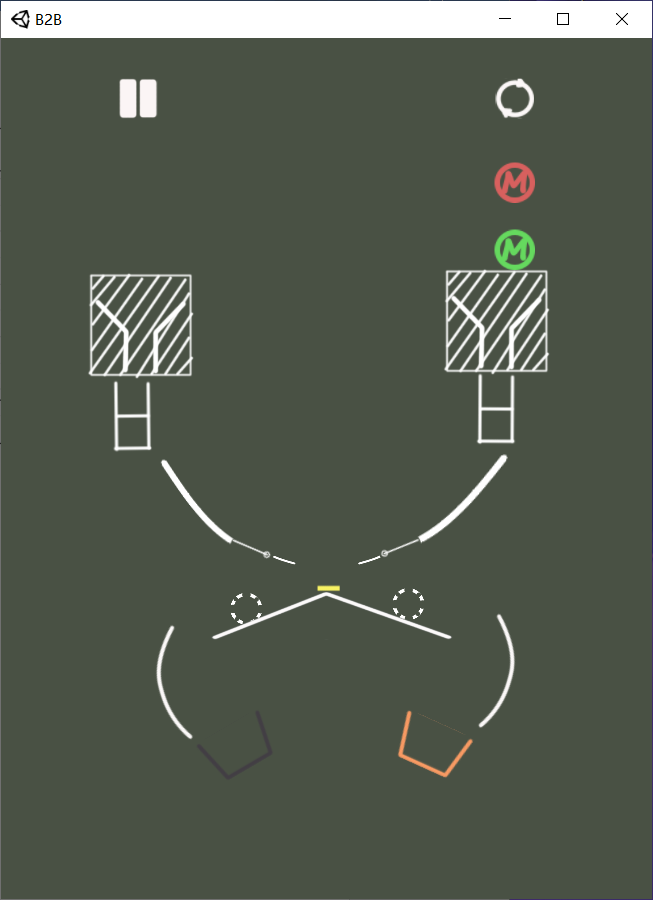
游戏攻略：入口：红→入口：任意→入口：绿

* + 1. Level 10



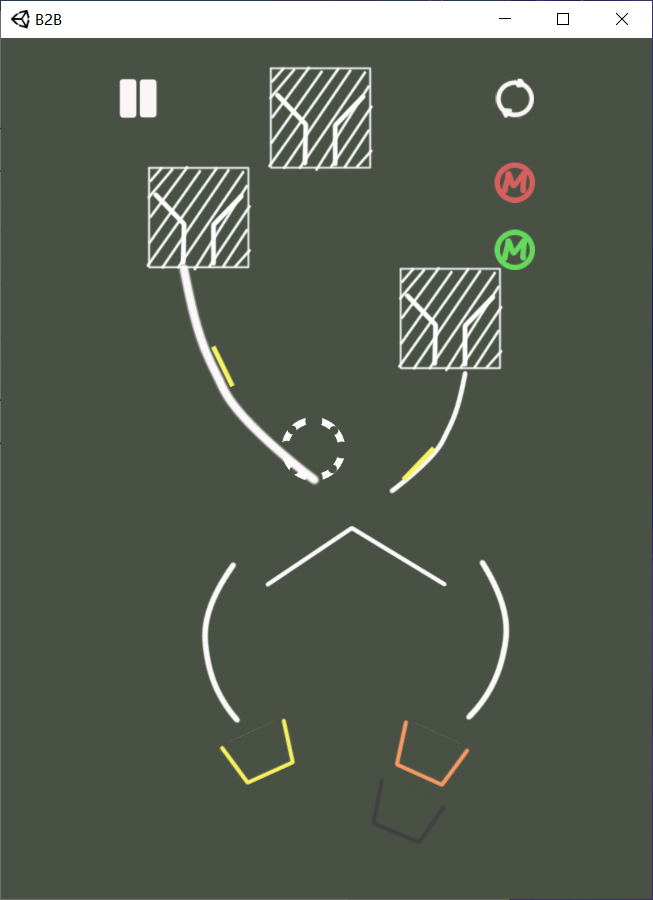
游戏攻略：右入口：红→右入口：红→右入口：红→右入口：蓝

* + 1. Level 11



游戏攻略：左入口：绿→左入口：绿→右入口：红→右入口：红

* + 1. Level 12



游戏攻略：右入口：任意→左入口：任意→中入口：绿→左入口：任意→中入口：红

# **总结（Conclusion）**

在这段超过三个月的时间里，我们组的成员们都在这款游戏上花了很多时间和精力。我们在这个过程中一起进步，一起学习。从开始项目的选题到编译环境的选择，我们都给出了自己的意见，并且达成了共识。我们成功地实现了暂停、重玩、小球生成、小球拖拽、小球销毁、生成物理小球、小球轨道设置、篮子承接、胜利失败判定、计时器与积分、开关、篮子堆叠、翻斗、染色器与混色器、加速器、退出游戏和切换界面等功能。 从一开始对Unity2D的学习，再到后来对游戏界面的搭建，对游戏物体道具的增加，我们都遇到了很多大大小小的问题，但是我们耐心地一点点地调试与重整，最终完成了项目。我们能够在这么短的时间内取得进步，是因为我们经常在一起讨论和交流想法。在交流过程中不断更新和成熟我们的程序代码，使我们的游戏运行得更加顺利，内容更加丰富。我们使用unity开发了一个有关小球运动与碰撞的物理休闲益智类游戏，并在其中加入诸如混色、叠加等设计增加其可玩性，再配以设计的烧脑关卡达到寓教于乐、在游戏中锻炼逻辑的目的，取名为B2B（Ball to Basket）。这款游戏的难度适中，轻松解压，对运行条件要求较低，适合大家在业余时间体验。总的来说，我们通过这一学期对 C 语言课程的学习和小组项目的制作，掌握了C语言和C#语言的基本内容和语法，并能够根据个人需求灵活使用来实现预期效果，我们的编程能力和程序设计能力在课程与项目中大大提高！