

hw2 2D-Game-with-Physics 游戏说明

功能概述

1. 背景 (10%)：实现可以移动的背景（详见exe文件）
2. 平台 (20%)：实现教程中提到的三种平台



Basic Platform

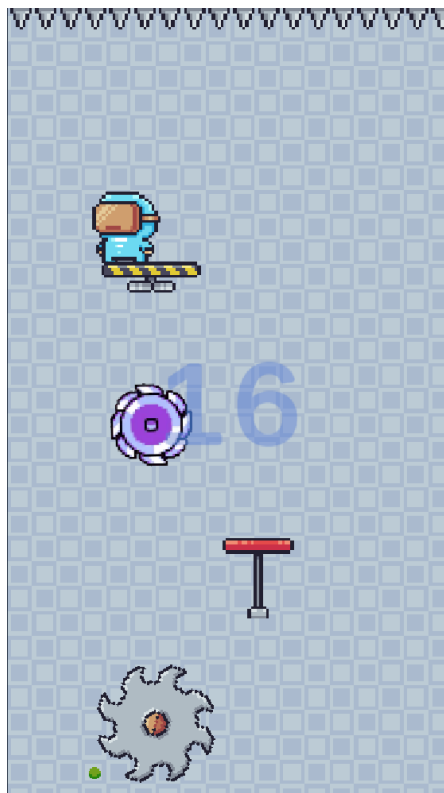


Rotate Platform



Spiked Ball

3. 角色 (15%)：实现角色移动（详见exe文件）
4. 生成 (20%)：实现无限平台生成和销毁



5. UI/游戏管理 (20%)：实现计分；实现游戏管理脚本



计分



计分



游戏管理

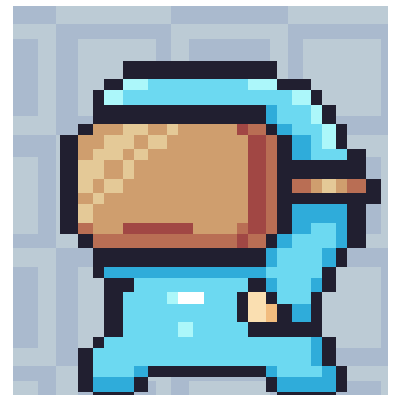
6. 角色动画 (15%)：实现角色动画切换



fall

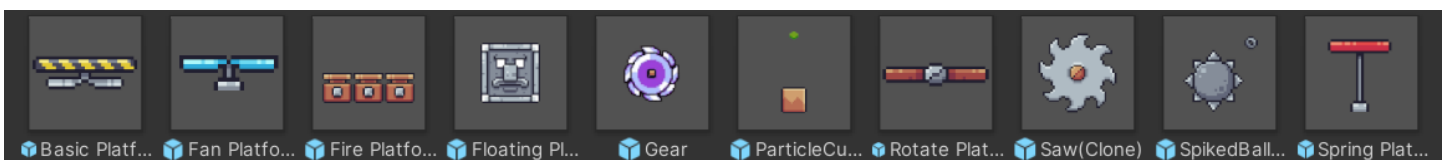


idle



walk

7. 加分项(+10%)：实现不同平台或障碍物，其中需要用到其它的物理组件或编写脚本实现平台特效等，可以参考原作者的视频教程实现风扇弹跳机关，也可以参照其他平台跳跃游戏实现其他平台或障碍。实现 其他对游戏机制的改进，也将根据实现难度和创意给予附加分



Trap总览

- 新增平台 Spring Platform
- 新增平台 FLoating Platform
- 新增平台 Particle Cube
- 新增平台 Fan Platform
- 新增平台 Gear

- 新增平台 Fire Platform
- 新增平台 Saw

新增平台 Spring Platform



Spring Platform



Spring Platform

1. 功能：

角色碰到Spring Platform后会被向上弹起，类似于蹦床

2. 实现方式：

- spring platform绑定Box Collider 2D
- 为spring platform添加“Spring”标签。PlayerController脚本中，检测到collision后，判断是否为“Spring”，若是在脚本中修改角色y轴方向速度
- 为spring platform添加Spring_idle和Spring_bounce动画，Spring_idle->Spring_bounce添加条件：名为“bounce”的Trigger，若与Player碰撞触发该Trigger。Spring_bounce->Spring_idle添加transition为solo，选择has exit time，时间为播放Spring_bounce动画一次的长度。

新增平台 Floating Platform



1. 功能

Player跳上去之后Rock Head会被压扁眨眼，平台上升速度会变慢

2. 实现方式

- 绑定spring joint 2d，固定z轴旋转、x轴位移
- 创建静止动画和撞击动画
- 同理Spring Platform创建animator状态机图、transition、trigger

新增平台 Particle Cube



Particle Cube

1. 功能

cube有粒子效果，与绿点有弹簧绑定

2. 实现方式

- Cube绑定box collider 2d
- 绿点rigidbody设置为static不受重力影响
- cube绑定spring joint 2d，固定点设置为图中绿点
- 调整spring参数

新增平台 Fan Platform



Fan Platform

1. 功能

Player粘上去后Fan开始旋转

2. 实现方式

- 检测与Player的碰撞
- 碰撞后播放旋转动画

新增平台 Gear



Gear

1. 功能

带有初角速度的齿轮，Player与其交互可改变旋转速度

2. 实现方式

- 创建Chain和Wheel
- Chain的rigidbody设置为static
- Wheel绑定Hinge Joint 2D，铰点设置为Chain
- Wheel绑定Circle Collider 2D
- Wheel绑定设置初始角速度的脚本

新增平台 Fire Platform



Fire Platform

1. 功能

跳上去之后就会死亡

2. 实现方式

- 创建empty object，将三个独立的平台作为子对象
- 创建燃烧动画
- 标签都设置为Spike

新增平台 Saw



Saw

1. 功能

具有初始角速度的锯齿，碰到即为死亡

2. 实现方式

- 创建Chain和Serration
- Chain的rigidbody设置为static
- Serration绑定Hinge Joint 2D，铰点设置为Chain
- Serration绑定Circle Collider 2D
- Serration绑定设置初始角速度的脚本
- Serration标签设置为Spike