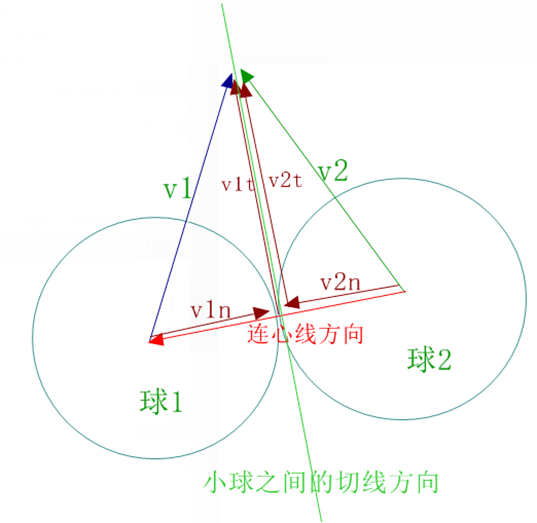
算法分析

一 逻辑原理：

移动水果，并进行边界检测计算出水果之间碰撞检测，如果是两个相同的水果，进行水果合并，否则计算小球碰撞后的移动方向。如果水果超出终点线，就将全部的水果粉碎，弹出结束页面，等待进入下一局。

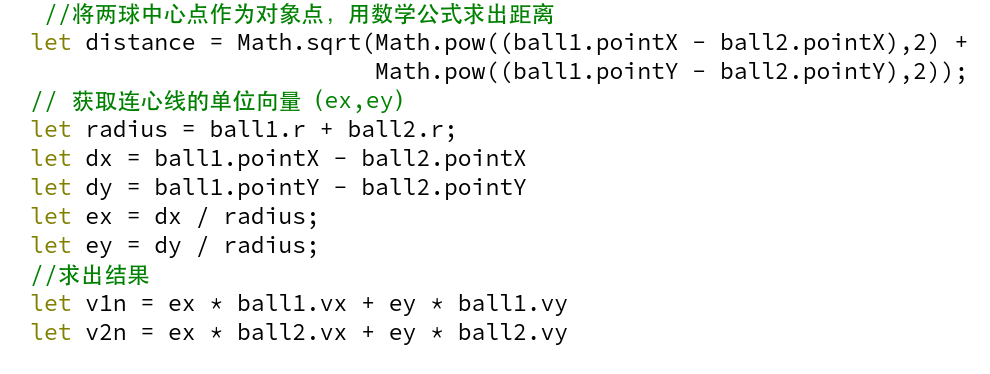
二 水果碰撞情景分析：

由于两个小球碰撞，切线上的速度都是互相平行的，没有作用力(如下图所示)。而连心线上是相互碰撞的，会有作用力，所以需要求出球1和球2的连心线方向上的速度值。然后再根据动量守恒定律和机械能守恒定律求出碰撞后的球1和球2的连心线方向。最后再互相加上各自在切线上的速度即可得到各自碰撞后的x速度，y速度。

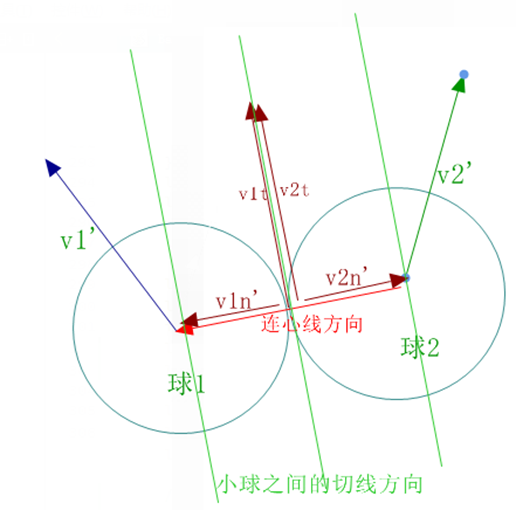


三 获取v1n和v2n

代码如图：



三 计算碰撞后的速度方向



v1' : 球1碰撞后的速度方向,等于v1n' + v1t

v2 ' : 球2碰撞后的的速度方向,等于v2n' + v2t

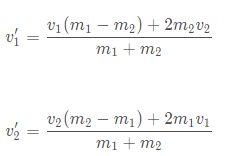
v1n'和v2n' : 两个小球碰撞后的投影速度

假如这两个小球是一样大, v1n'和v2n'取值就是:

v1n' = v2n

v2n' = v1n

并且根据动量守恒定律和机械能守恒定律得出:

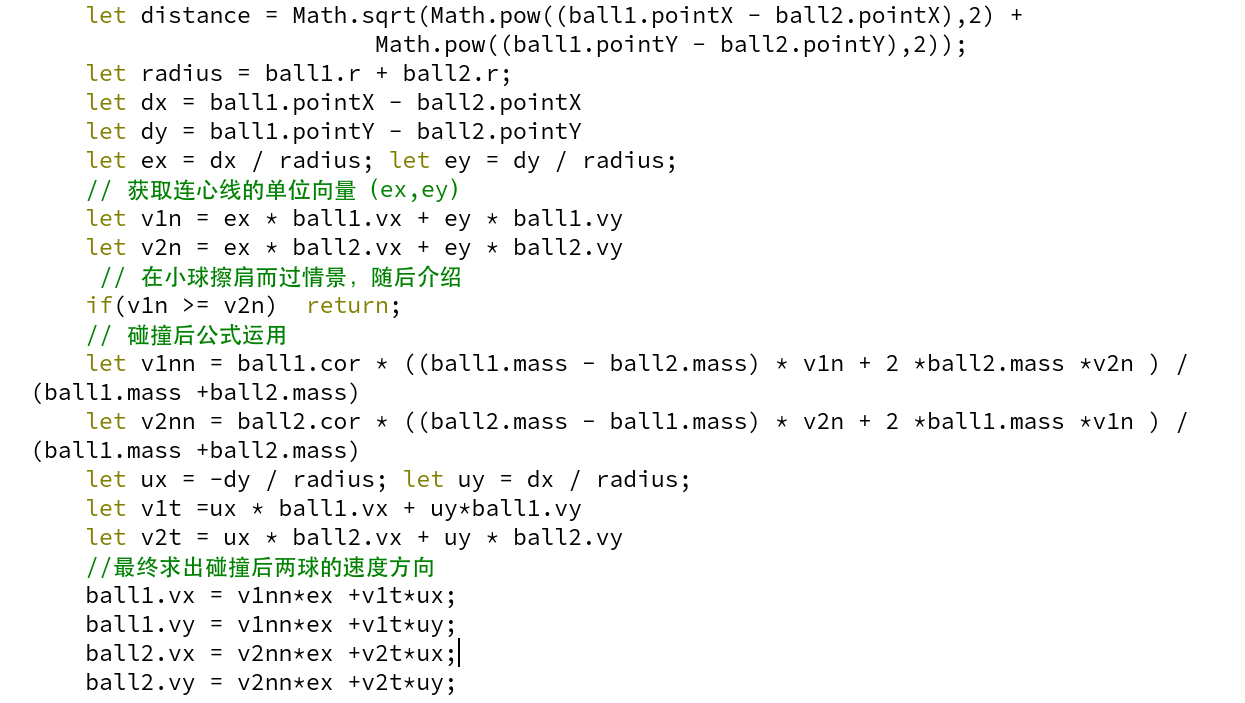


v1和v2 : 两个小球碰前速度.

m1和m2 : 两个小球的质量

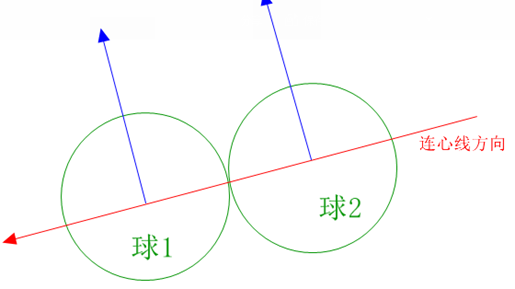
v1'和v2' : 两个小球碰后速度

代码如图：



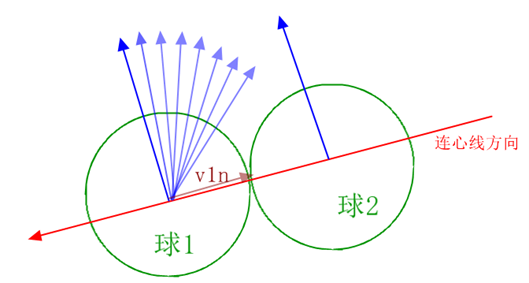
四：小球擦肩而过情景

两个小球平行移动场景图:



如球1和球2在平行移动,那么他们与连心线的夹角恰好是90°, v1n和v2n则都为0

假如球1的夹角大于了球2的夹角,那么就会出现碰撞,如下图所示:



虚线箭头速度方向表示球1的夹角大于球2的夹角的时候场景.

而cos的取值方式刚好是在0~180°的时候,角度越大,值越小,所以v1n >=v2n时,则不会碰撞.

五：小球一直降落在小球的正上方（即垂直降落）

这时候,小球由于没有切线上的速度方向,所以在重力加速度下,会慢慢让小球们堆起来,从而游戏结束.

所以还需要在碰撞后末尾添加以下判断:

