类图：



Movable

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| model | Actor | 模型 |
| speed | int | 每一帧可以移动的距离 |
| direction | int | 面向的方向，相对于相机的y轴正方向的逆时针角度 |
| position |  | 模型在场景3维坐标系下的位置 |
| colliderNodePath | NodePath | 碰撞体的节点路径 |
| colliderName | string | 碰撞体的名称 |
| sounds | map<string,sound> | 音效映射：move、attack、underAttack |

主角

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| maxHP | int | 主角最大生命值 |
| HP | int | 主角当前生命值，当其为0时主角死亡，可通过道具进行回复 |
| attackMethod | int | 主角的攻击方式，用于在主角攻击时，设定主角的攻击特性 |
| attackPower | int | 主角的攻击力，在受攻击者的防御为0时，受到的伤害值 |
| animations | map<string,string> | 主角动画映射：move、attack、underAttack |
| isUnderAttack | int | 指示主角是否处于受攻击状态 |

主角移动

|  |  |
| --- | --- |
| 函数签名 | public move(self) |
| 函数功能 | 处理玩家发送移动指令事件   1. 判断 “主角是否处于受攻击状态” 是否为真，如果为真，直接返回； 2. 根据主角的朝向、位置、速度来更新主角的位置； 3. 同时播放”主角移动“动画； 4. 向音效管理器发送“主角移动”事件； |
| 输入参数 | self:类对象引用 |
| 返回结果 | 无 |

主角攻击

|  |  |
| --- | --- |
| 函数签名 | public attack(self) |
| 函数功能 | 处理玩家发送攻击指令事件   1. 判断 “主角是否处于受攻击状态” 是否为真，如果为真，直接返回； 2. 根据主角的朝向、位置、攻击力，初始化子弹的攻击力、初始化位置、朝向； 3. 根据主角攻击方式，设置子弹的来源、数量、体积倍数、朝向、速度； 4. 向房间发送”子弹生成“事件（负责将子弹加入碰撞检测事件处理） |
| 输入参数 | self:类对象引用 |
| 返回结果 | 无 |

主角受到攻击

|  |  |
| --- | --- |
| 函数签名 | public underAttack(self,collisionEntry) |
| 函数功能 | 处理主角受到攻击（近战或远程）事件   1. 判断 “主角是否处于受攻击状态” 是否为真，如果为真，直接返回； 2. 根据attackPower决定主角受到的伤害值，相应地减少主角的当前生命值； 3. 设置 “主角是否处于受攻击状态” 为真； 4. 如果是远程攻击，，并且子弹为”不穿透“类型，向房间发送”子弹攻击“事件（以销毁子弹） 5. 如果主角当前生命值小于等于0，播放”主角死亡“动画，完毕后，向关卡管理器发送”主角死亡“事件 |
| 输入参数 | self:类对象引用；collisionEntry：攻击者 |
| 返回结果 | 无 |

主角拾取道具

|  |  |
| --- | --- |
| 函数签名 | public fetchItem(self,collisionEntry) |
| 函数功能 | 处理主角碰撞道具事件   1. 根据道具功效，提升主角属性 |
| 输入参数 | self:类对象引用；collisionEntry：道具 |
| 返回结果 | 无 |

主角改变朝向

|  |  |
| --- | --- |
| 函数签名 | public updateDirection(self) |
| 函数功能 | 主循环任务   1. 在每一帧中，根据鼠标位置，更新主角朝向 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

怪物

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| index | int | 怪物在房间中的编号 |
| maxHP | int | 怪物最大生命值 |
| HP | int | 怪物当前生命值，当其为0时主角死亡，可通过道具进行回复 |
| attackMethod | int | 怪物的攻击方式，用于在怪物攻击时，设定怪物的攻击特性 |
| attackPower | int | 怪物的攻击力，在受攻击者的防御为0时，受到的伤害值 |
| animations | map<string,string> | 怪物动画映射：move、attack、underAttack |
| isUnderAttack | int | 指示怪物是否处于受攻击状态 |
| isBoss | int | 指示怪物是否为boss |
| isAttacking | int | 指示当前怪物是否处于攻击状态 |

怪物移动

|  |  |
| --- | --- |
| 函数签名 | public move(self) |
| 函数功能 | 主循环任务   1. 判断 “是否处于受攻击状态” 是否为真，如果为真，直接返回； 2. 判断“是否为boss”为真且 “是否处于攻击状态” 是否为真，如果为真，直接返回； 3. 更新AI 4. 同时播放”移动“怪物动画； 5. 向音效管理器发送“怪物移动”事件； |
| 输入参数 | self:类对象引用 |
| 返回结果 | 无 |

史莱姆和小机器人-大史莱姆和双生机器人-攻击

|  |  |
| --- | --- |
| 函数签名 | public attack(self，collisionEntry) |
| 函数功能 | 处理与主角的碰撞事件   1. 判断 “是否处于受攻击状态” 是否为真，如果为真，直接返回； 2. 近战攻击   近战攻击：  （主角处理碰撞检测事件，怪物不操作） |
| 输入参数 | self:类对象引用；collisionEntry：主角 |
| 返回结果 | 无 |

蝙蝠和飞行机器人-攻击

|  |  |
| --- | --- |
| 函数签名 | public attack(self，collisionEntry) |
| 函数功能 | 处理与主角的碰撞事件   1. 判断 “是否处于受攻击状态” 是否为真，如果为真，直接返回； 2. 范围攻击   范围攻击：   1. 延迟一秒后，播放”自爆动画“ 2. 向房间发送”怪物死亡“事件 |
| 输入参数 | self:类对象引用；collisionEntry：主角 |
| 返回结果 | 无 |

机器头和骷髅头-攻击

|  |  |
| --- | --- |
| 函数签名 | public attack(self，collisionEntry) |
| 函数功能 | 处理与主角的碰撞事件   1. 判断 “是否处于受攻击状态” 是否为真，如果为真，直接返回； 2. 远程攻击   远程攻击：   1. 根据怪物的朝向、位置、攻击力，初始化子弹的攻击力、初始化位置、朝向； 2. 根据怪物攻击方式，设置子弹的来源、数量、体积倍数、朝向、速度； 3. 将子弹列表发送到当前房间对象（负责将子弹加入碰撞检测事件处理） |
| 输入参数 | self:类对象引用；collisionEntry：主角 |
| 返回结果 | 无 |

魔法柱和电磁塔-攻击

|  |  |
| --- | --- |
| 函数签名 | public attack(self) |
| 函数功能 | 主循环任务  第一次初始化：   1. 初始化子弹列表，设定子弹为”穿透”类型 2. 根据怪物攻击方式，设置子弹的来源、数量、体积倍数、朝向、速度、初始位置； 3. 将子弹列表发送到当前房间对象（负责将子弹加入碰撞检测事件处理）   攻击：  1、不断更新子弹的朝向 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

毛毛虫和小火车-攻击

|  |  |
| --- | --- |
| 函数签名 | public attack(self) |
| 函数功能 | 主循环任务  第一次初始化：   1. 初始化子弹列表，设定子弹为”穿透”类型 2. 根据怪物攻击方式，设置子弹的来源、数量、体积倍数、朝向、速度、初始位置，设定速度为0 3. 将子弹列表发送到当前房间对象（负责将子弹加入碰撞检测事件处理）   攻击：  1、不断更新子弹range，当range为0时，向房间发送“子弹销毁”事件 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

牛头幽灵和机械蜘蛛-攻击

|  |  |
| --- | --- |
| 函数签名 | public attack(self) |
| 函数功能 | 主循环事件   1. 每五秒随机触发一种攻击 2. 设置‘是否处于攻击状态’为真 3. 具体攻击方式 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

牛头幽灵

技能1：冲撞

|  |  |
| --- | --- |
| 函数签名 | public bump(self) |
| 函数功能 | 1. 设置‘是否处于攻击状态’为真 2. 存储主角的位置pos 3. 延迟一秒后，向pos方向，加速移动，维持0.3秒后，恢复速度 4. 设置‘是否处于攻击状态’为假 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

技能2：锥形火焰

|  |  |
| --- | --- |
| 函数签名 | public Conical\_flame(self) |
| 函数功能 | 1. 设置‘是否处于攻击状态’为真 2. 存储主角的位置pos 3. 播放boss”Conical\_flame”动画 4. 延迟一秒后， 5. 初始化火焰模型，创建火焰碰撞体，设置火焰攻击力、位置、方向（boss到pos的向量），向房间发送“火焰生成”事件（以进行碰撞检测），存留2s后销毁，向房间发送“火焰销毁“事件 6. 设置‘是否处于攻击状态’为假 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

技能3：召唤

|  |  |
| --- | --- |
| 函数签名 | public summon(self) |
| 函数功能 | 1. 设置‘是否处于攻击状态’为真 2. 播放boss”summon”动画 3. 延迟0.5s后， 4. 初始化3个史莱姆模型，设置其初始位置，并向房间发送”怪物生成“事件 5. 设置‘是否处于攻击状态’为假 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

机械蜘蛛：

技能1：坠地攻击

|  |  |
| --- | --- |
| 函数签名 | public drop(self) |
| 函数功能 | 1. 设置‘是否处于攻击状态’为真 2. 存储主角的位置pos 3. 播放”升天“动画（0.5s） 4. 更新位置为pos 5. 播放”坠地动画“（0.5s） 6. 设置‘是否处于攻击状态’为假 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

技能2：全屏弹幕

|  |  |
| --- | --- |
| 函数签名 | public fullScreenBarrage(self) |
| 函数功能 | 1. 设置‘是否处于攻击状态’为真 2. 延迟1s 3. 生成全屏随机子弹（一圈又一圈），每隔1s，增加一次，向房间发送“子弹生成”事件，持续5s 4. 设置‘是否处于攻击状态’为假 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

技能3：治疗

|  |  |
| --- | --- |
| 函数签名 | public v(self) |
| 函数功能 | 1. 设置‘是否处于攻击状态’为真 2. 延迟1s后 3. 播放”恢复“动画（5s） 4. 每秒恢复怪物一定（10）生命值 5. 持续五秒 6. 设置‘是否处于攻击状态’为假 |
| 输入参数 | self:类对象引用； |
| 返回结果 | 无 |

怪物受到攻击

|  |  |
| --- | --- |
| 函数签名 | public underAttack(self,collisionEntry) |
| 函数功能 | 处理怪物受到攻击（近战或远程）事件   1. 判断 “是否处于受攻击状态” 是否为真，如果为真，直接返回； 2. 根据攻击者的attackPower决定怪物受到的伤害值，相应地减少怪物的当前生命值； 3. 设置 “怪物是否处于受攻击状态” 为真； 4. 向房间发送”子弹攻击“事件（以销毁子弹） 5. 如果普通怪物当前生命值小于等于0，播放”怪物死亡“动画，完毕后，向房间发送”怪物死亡“事件；而特定怪物当前生命值小于等于0，可以生成两个相同类型的普通怪物，并向房间发送”普通怪物生成“事件 |
| 输入参数 | self:类对象引用；collisionEntry：攻击者 |
| 返回结果 | 无 |

子弹

|  |  |  |
| --- | --- | --- |
| 属性 | 类型 | 描述 |
| from | string | 描述子弹来源：hero、ogre |
| type | int | 穿透、不穿透 |
| attackPower | int | 攻击力，在受攻击者的防御为0时，受到的伤害值 |
| range | int | 子弹在场景中最大的可移动距离 |
| volume | int | 子弹的体积（碰撞体积） |

子弹移动

|  |  |
| --- | --- |
| 函数签名 | public move(self) |
| 函数功能 | 主循环任务   1. 判断子弹的range是否为0，当为0时，向房间发送”子弹消失“事件 2. 根据子弹的朝向、移动速度，更新子弹在场景中的位置 |
| 输入参数 | self:类对象引用 |
| 返回结果 | 无 |