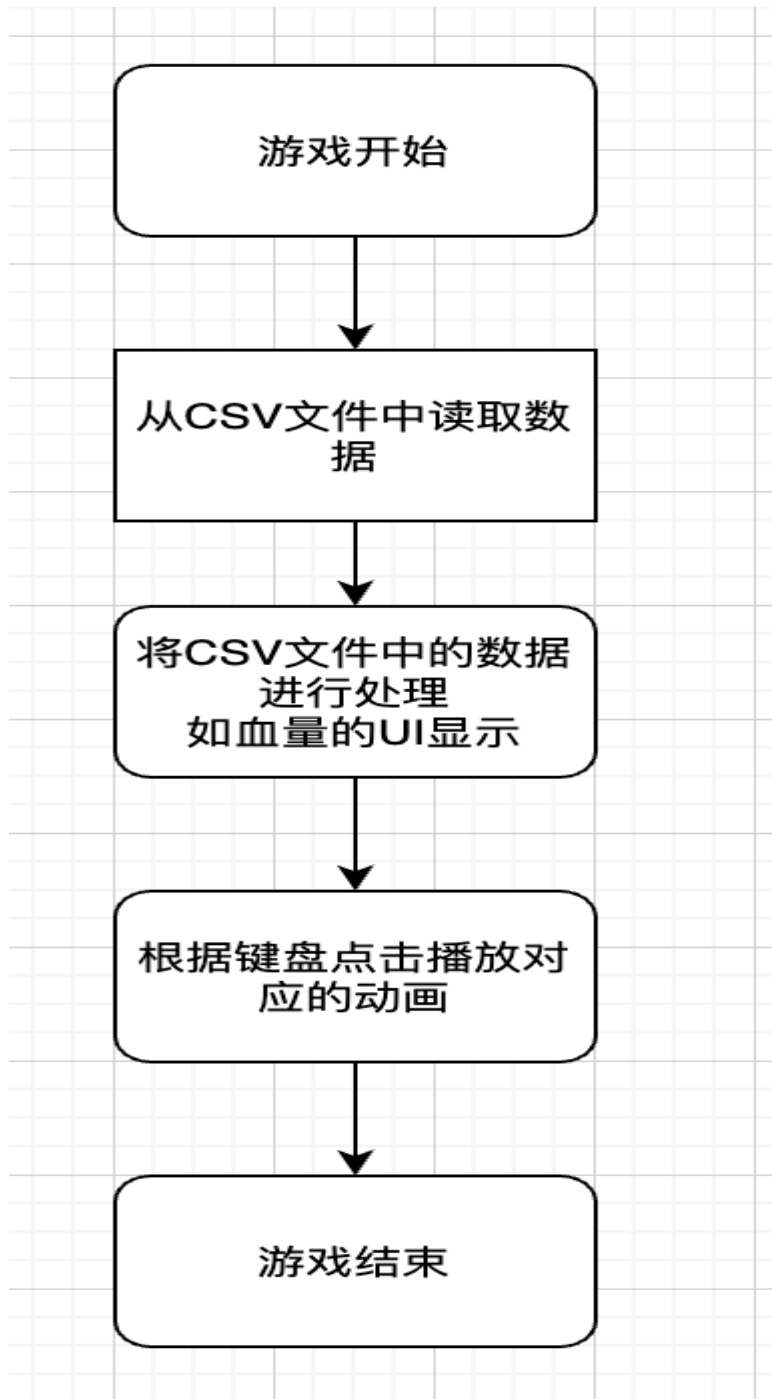


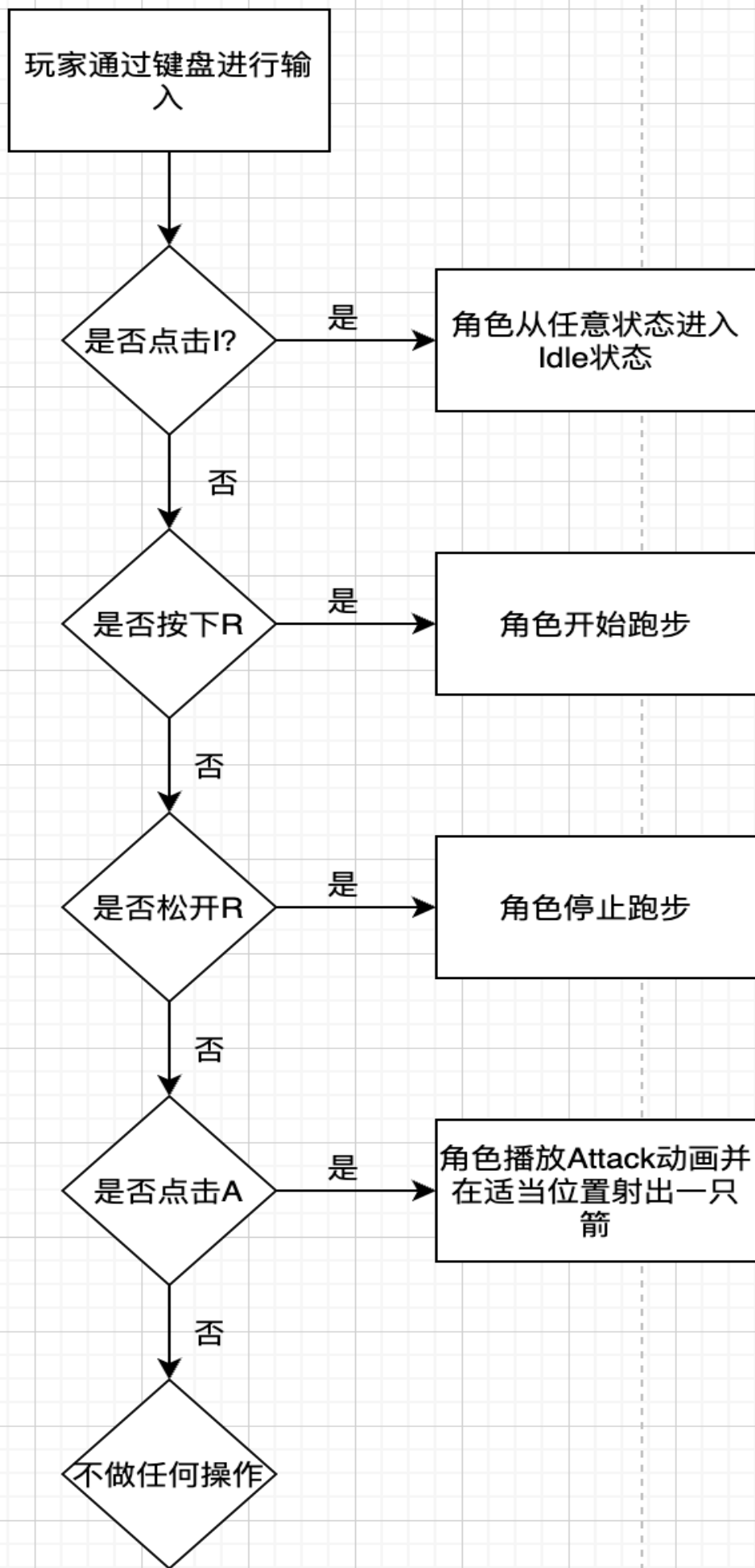
# 动画技术文档

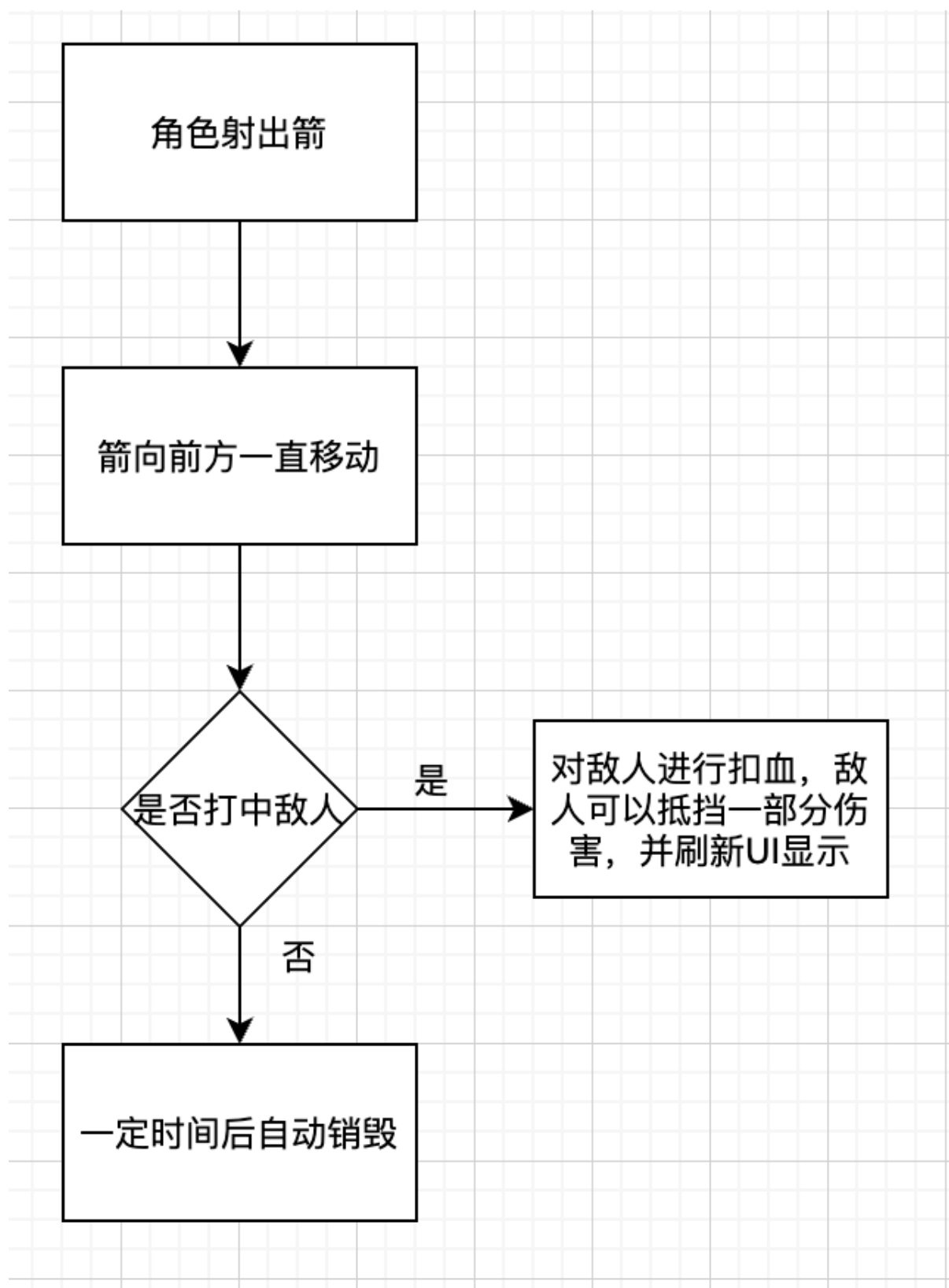
## 1. 整体框架

主要思路是通过Animator控制角色动画播放，使用碰撞器判断箭头碰撞扣血

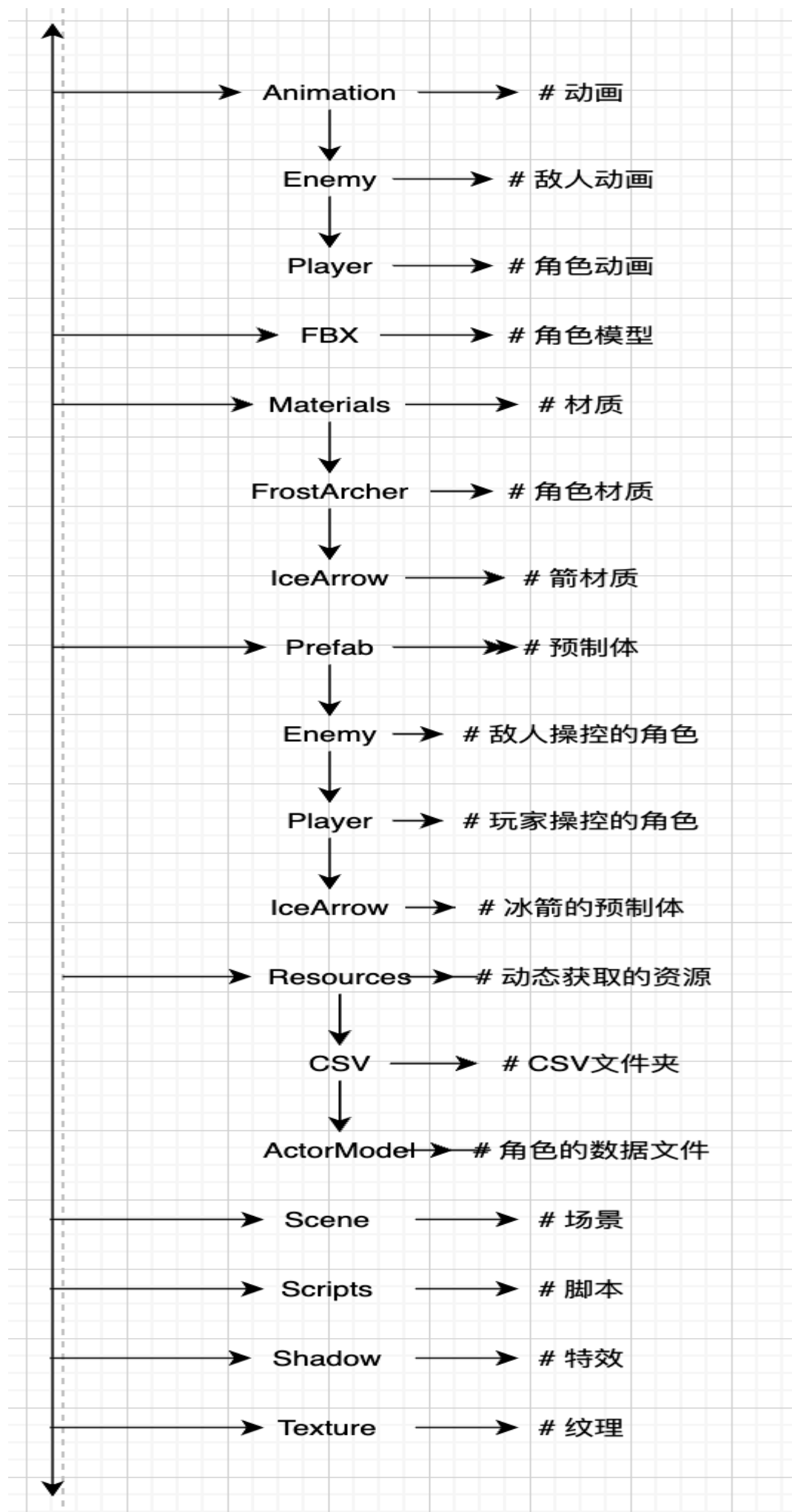
## 2. 流程图







### 3. 目录结构



#### 4. 代码逻辑分层

类名	分类	主要职责
ActorActive	Controller	Player和Enemy的共有基类，提供了两者共有的行为和属性
Arrow	Controller	控制箭的移动和销毁，包括箭的碰撞检测，碰撞到敌人就对敌人进行扣血
EnemyController	Controller	敌人控制类，控制敌人的动画播放和扣血等行为
PlayerController	Controller	玩家控制类，控制角色的动画播放和攻击等行为
ActorModel	Model	Player和Enemy的信息类，提供了两者的数据信息，从CSV文件中动态读取
ActorUIMessageShow	View	UI控制类，Player和Enemy共用，提供了刷新血量的UI显示

#### 5. 关卡结构

场景中主要的物体有Ground，GameModelManager，IceArrows，Player，Enemy

Ground: 场景中的地板

GameModelManager: 提供GameModelManager脚本，用于获取CSV文件解析后的数据

IceArrows: 用于收纳创建的箭

Player: 玩家控制的角色

Enemy: 敌人控制的角色

Player与Enemy的物体基本一样：

由FrostArcher和Canvas组成：

FrostArcher是FBX文件，提供角色的模型

Canvas用来显示角色的血条，主要有文本显示和滑动条显示