**森林冰火人**

**一、项目概述**

**1.项目背景**

**1.1项目开发的目的和目标**

项目的目的是开发一个名为“森林冰火人”的游戏，通过游戏提供娱乐和挑战，吸引玩家的注意力，同时也锻炼玩家的反应能力和解决问题的能力。此游戏将呈现冰娃和火娃两个角色的冒险故事，让玩家在不同的关卡中控制角色穿越各种障碍和谜题。

**1.2技术选择和预期效果**

（1）编程语言和图形库：本项目使用C语言，并结合EasyX图形库。

（2）预期效果：

a.娱乐性：游戏的主要目标是为玩家提供娱乐和消遣，使他们在游戏过程中感到愉快。

b.培养技能：游戏可能培养玩家的空间认知、反应能力、逻辑思维等。

c.用户互动：通过键盘输入控制角色，玩家可以与游戏进行互动，体验游戏中的交互性。

d.图形展示：图形库可以实现图像的绘制和展示，让游戏画面更具吸引力。

**2.项目简介**

“森林冰火人”是一个横向冒险平台游戏，玩家需要控制冰娃和火娃在各种地图中完成任务。游戏的主要操作包括横向移动、纵向跳跃以及切换角色。游戏地图中包含地面、墙壁、冰水、火水和毒液等元素，冰娃和火娃需要躲避危险并合作解决难题。

**二、需求分析**

**1.功能需求**

**1.1基本功能需求**

（1）游戏版面与地图设计：游戏需要一个可视化的游戏界面，包括地图、角色、障碍物和元素等。地图应该具有不同的难度和关卡设计。设计游戏窗口，定义游戏画面尺寸和分辨率。绘制游戏地图，包括地面、墙壁、冰水、火水、毒水和钻石等元素。

（2）地图表示与初始化：游戏应该能够根据设计好的地图数据来绘制游戏场景，并在游戏开始时初始化地图，放置角色和元素。

（3）冰火娃动作图片切换：设计冰娃和火娃的角色模型，可以使用图片表示角色的不同状态。根据玩家输入和角色状态，游戏需要能够实时切换冰娃和火娃的动作图片，包括不同的姿势、移动状态和跳跃状态。创建角色数据结构，包含位置、状态、动画帧等信息。

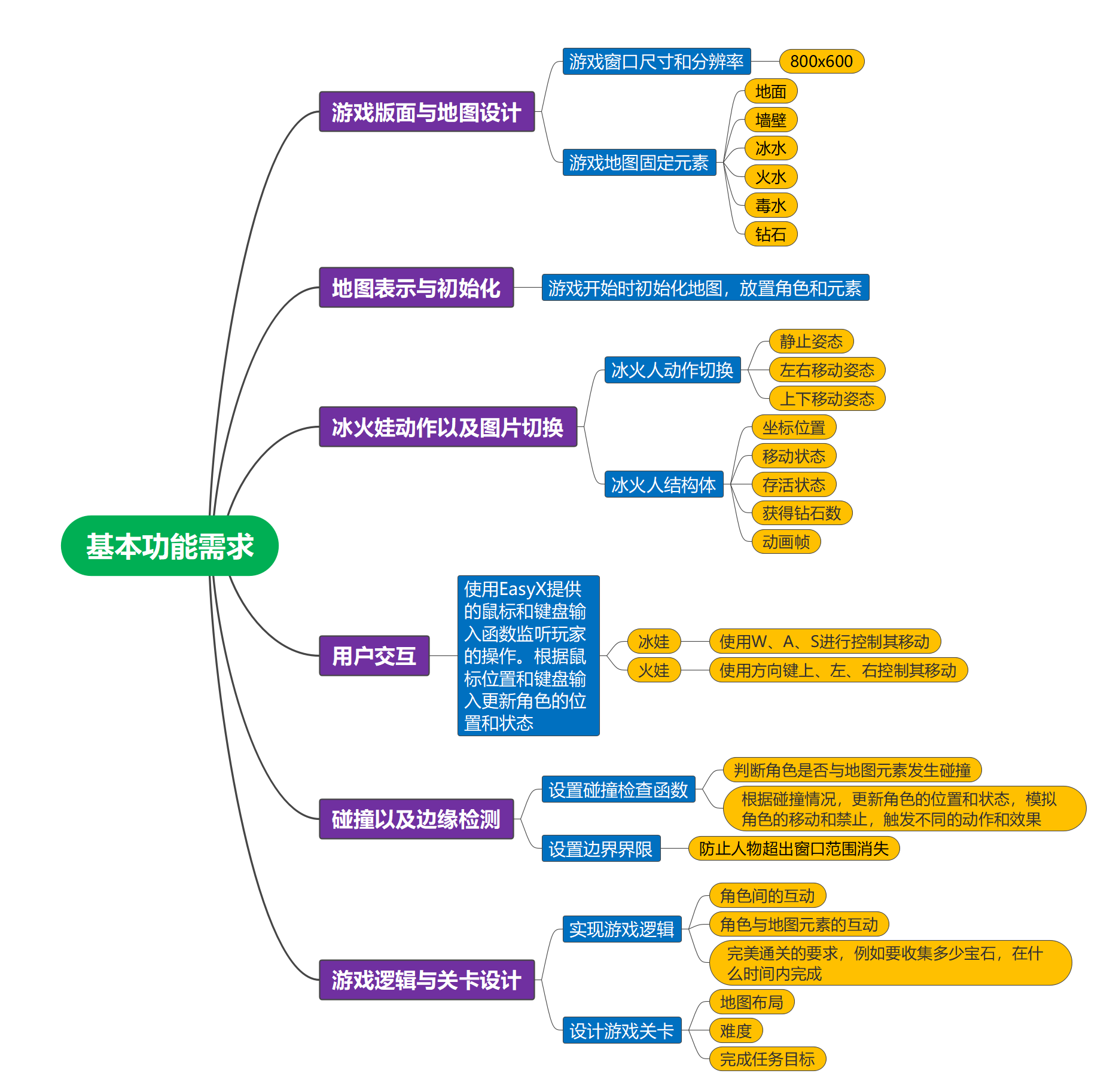
（4）用户交互与输入处理：使用EasyX提供的鼠标和键盘输入函数，监听玩家的操作。根据鼠标位置和键盘输入更新角色的位置和状态。

（5）碰撞检测与物理模拟：实现碰撞检测函数，判断角色是否与地图元素发生碰撞。根据碰撞情况，更新角色的位置和状态，模拟角色的移动和跳跃。

（6）冰火娃的跑动：玩家可以通过移动鼠标来控制冰娃和火娃的横向移动，移动到屏幕边缘会改变角色的方向。同时，玩家通过点击左键控制角色的纵向移动，实现跳跃动作。玩家可以通过操控键盘来控制冰娃和火娃的移动，使用左右箭头键和A、D键控制冰娃和火娃的横向移动，使用上下箭头键和W、S键来控制冰娃和火娃的纵向移动跳跃。

（7）地面、墙壁、冰水、火水与毒液判断：游戏需要实现碰撞检测功能，以判断角色是否与地面、墙壁、冰水、火水、毒液等产生碰撞，从而触发不同的动作和效果。

（8）游戏逻辑与关卡设计：设计游戏关卡，包括不同的地图布局、难度和目标。实现游戏逻辑，处理角色之间的互动以及角色与地图元素的互动。



基本功能需求

**1.2.拓展功能需求**

(1)进阶版地图布局与提示功能：为增加游戏的难度和趣味性，可以设计更加复杂的地图布局，加入各种谜题和障碍。同时，提供提示功能，让玩家在困难关卡中获得帮助。

(2)自主设计游戏功能

a.角色能力与道具：为冰娃和火娃添加特殊能力，例如冰娃可以冻结冰水，火娃可以点燃木块等。引入道具系统，让玩家能够获得临时性的加强能力或帮助。

b.时间挑战与限时任务：在某些关卡中，加入时间限制，让玩家在有限的时间内完成任务。 设计限时任务，玩家需要在规定时间内完成特定的目标。

c.故事模式与剧情设计：设计游戏的故事背景，为游戏增加更多的情感和情节。引入剧情设计，通过动画或对话展现角色之间的互动。

d.反重力与特殊物理效果：设计反重力关卡，玩家需要应对不同的物理环境和效果。引入特殊物理效果，如悬浮、弹性等，为游戏增加趣味性。

**2.非功能需求**

2.1性能需求

（1）响应时间：游戏操作应该有足够的响应时间，确保玩家的输入能够迅速影响游戏画面。

（2）流畅性：游戏画面应该保持流畅，以提供良好的视觉体验。

2.2可靠性需求

（1）稳定性：游戏应该稳定运行，不应该因为异常情况而崩溃或产生严重错误。

（2）备份和恢复：在可能的情况下，游戏应该支持保存游戏进度，以便玩家在之后能够恢复游戏。

2.3安全性需求

（1）用户隐私：游戏不应该获取或共享用户的敏感信息，以保护用户的隐私。

（2）防作弊：游戏应该具备一定的防作弊机制，确保玩家在游戏中获得公平的体验。

2.4用户界面需求

（1）易用性：游戏界面应该简洁明了，易于理解和操作，适合不同年龄段的玩家。

（2）可定制性：如果有设置选项，应该允许玩家根据个人喜好进行一些界面设置。

2.5设计约束

（1）图像和音效：游戏图像和音效应该是原创或经过合法授权的，不应该侵犯他人的知识产权。

（2）资源占用：游戏在计算机上的资源占用应该合理，不应过于占用内存、处理器等资源。

**三、设计方案**

**1.数据描述**

**1.1地图元素对象的定义**

（1）地图元素对象表示游戏地图中的各种元素，包括地面、墙壁、钻石、火水、冰水、毒水等。

（2）定义其结构体包括：

typedef struct {

int x, y; // 地图元素的坐标

int width, height; // 地图元素的宽度和高度

int type; // 地图元素类型，如地面、墙壁、水、火等

IMAGE image; // 地图元素图像

} MapElement;

**1.2道具对象的定义**

（1）道具对象表示游戏中的各种道具，玩家之间可以与之交互，共同协作通关，例如游戏中的横梯子机关、方形石头等。

（2）定义其结构体包括：

typedef struct {

int x, y; // 道具的坐标

int width, height; // 道具的宽度和高度

int type; // 道具类型

IMAGE image; // 道具图像

} Item;

**1.3角色对象的定义**

（1）角色对象表示玩家在游戏中操控的两个游戏人物，冰娃和火娃。

（2）定义结构体包括：

typedef struct {

int x, y; // 角色坐标

int width, height; // 角色宽度和高度

int velocityX, velocityY; // 角色在X和Y方向上的速度

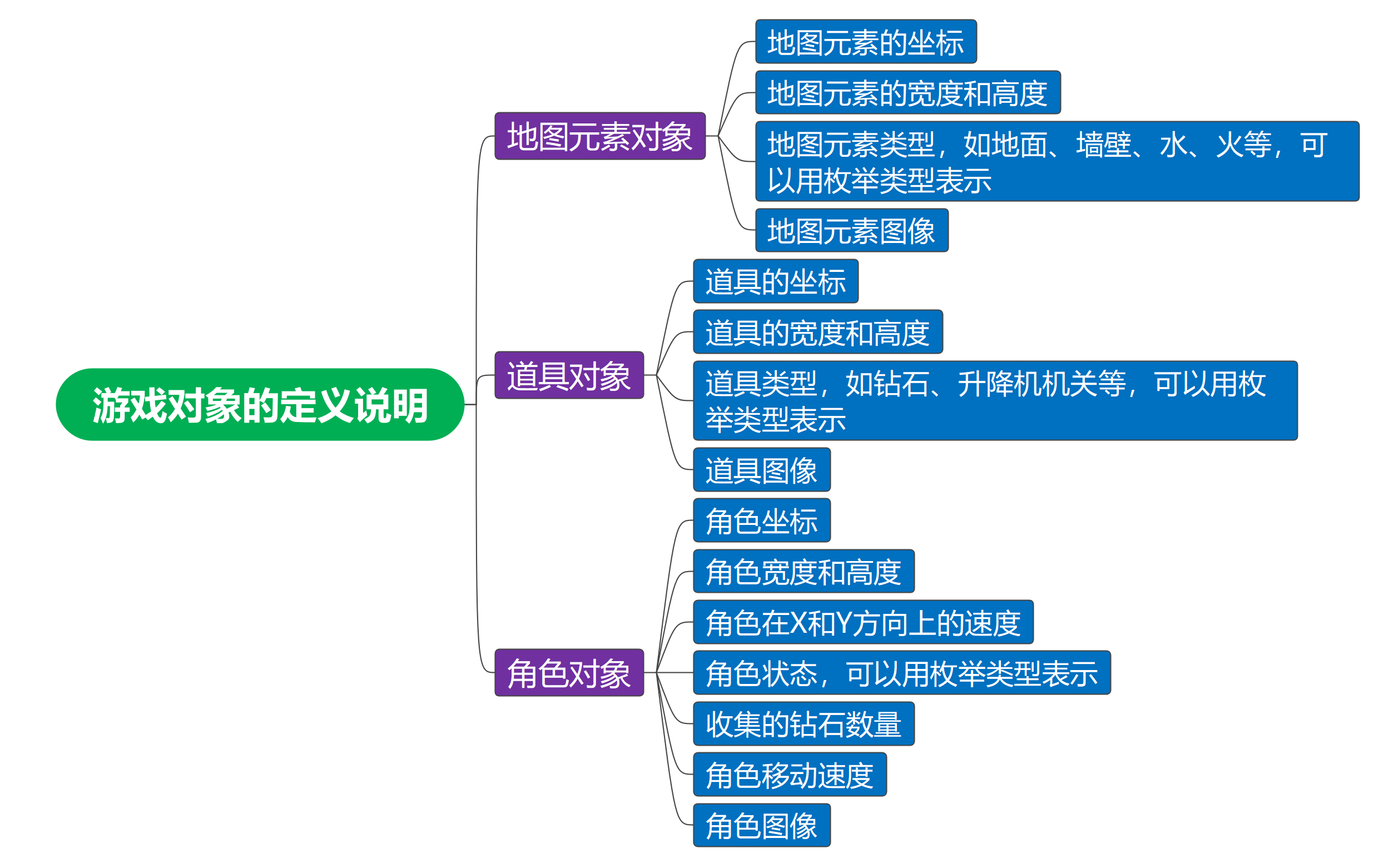
int status; // 角色状态

int diamonds; // 收集的钻石数量

int speed; // 角色移动速度

IMAGE image; // 角色图像

} Character;



游戏对象的定义说明

**2.功能设计**

**2.1角色移动和跳跃**

（1）通过监测键盘输入，控制角色的移动和跳跃。根据角色当前状态（是否在地面上）或者根据角色与地图的某些边缘坐标是否相交来判断角色是否能跳跃或者会不会落下，通过调整垂直速度来实现跳跃效果以及下落效果。

（2）在角色的更新循环中，更新角色的位置和速度以及图片，并根据地图元素的碰撞情况来调整角色的位置和状态，根据角色所前进的方向实时更新所对应的方向的图片以此来实现动态效果。

**2.2碰撞检测与地图元素交互**

（1）实现碰撞检测函数，判断角色是否与地图元素相交。

（2）当角色与地图元素发生碰撞时，根据地图元素的类型进行不同处理，如站在地面上、碰到墙壁，不同角色碰到所对应的钻石所界面的变化，以及角色收集钻石数的变化等。

（3）确保角色在合法位置移动，设置边界，避免随意穿越墙壁或地面。

（4）实现游戏逻辑，处理角色之间的互动和角色与地图元素的互动。

（5）用结构体数组存储地面，墙壁，各种机关的坐标（坐标是在游戏关卡界面中实际测出来的）当遇到地面坐标，向下速度减小为0。

（6）当遇到上方挡板，向上的速度减小为0。

（7）当遇到墙面坐标，向左/右的速度减小为0。

（8）当遇到冰水，火水与毒液，判断角色死亡。

（9）当人物坐标与机关坐标重合，则机关缓缓开启。

（10）当冰火娃坐标在出口门的范围内，门缓缓打开，游戏通关。

**2.3游戏界面与地图角色设计**

（1）创建游戏窗口，定义窗口大小和标题。

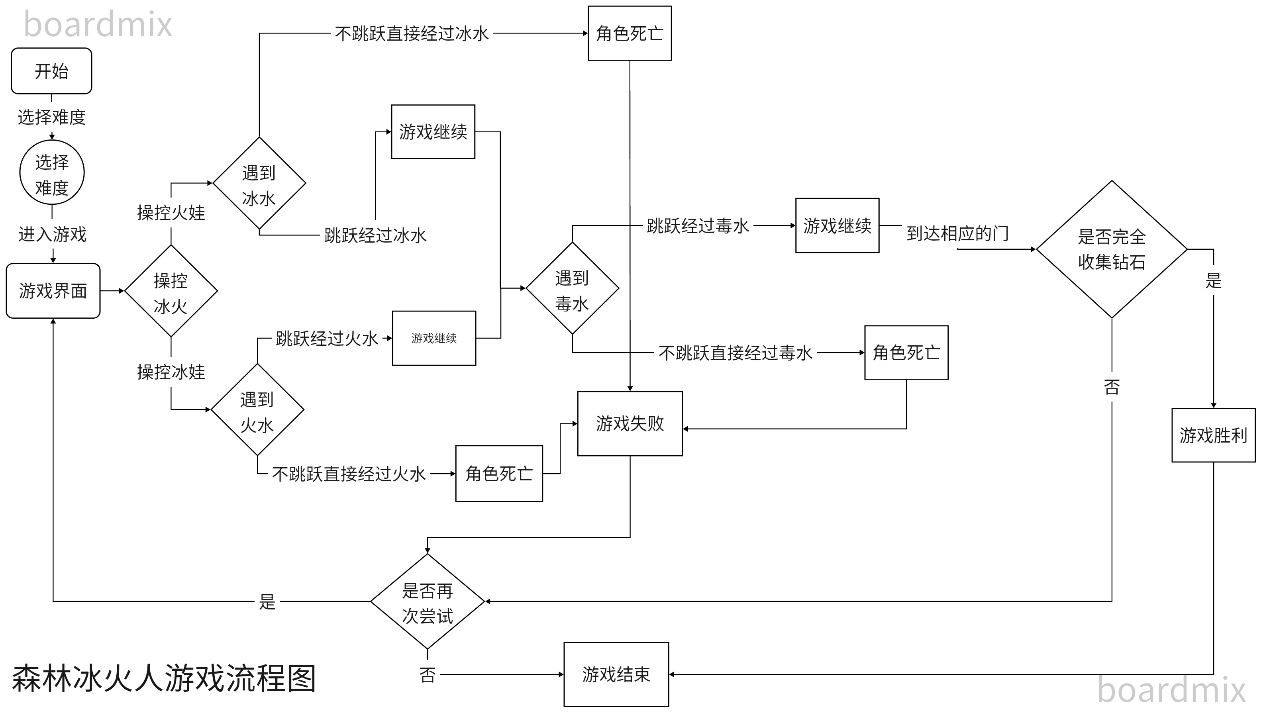
（2）在窗口中绘制游戏地图，使用不同颜色或图片表示不同地图元素。通过寻找素材来绘制地图，并在地图上各个特定位置贴上不同的地图元素，例如、冰水、火水、毒液和宝石等。

（3）使用图片资源表示冰娃和火娃的不同动作状态，如站立、跑动、跳跃等。

（4）定义两个角色数据结构体，包括坐标位置、状态和动画帧计数等。

**2.4用户交互与输入处理**

使用EasyX提供的函数监听鼠标和键盘输入，根据输入更新角色的位置和状态。当键入相应的方向时，冰火娃的运动状态也会发生改变，开始先由静止获得一定的加速度，然后速度大到一定的程度时保持匀速，松开按键后迅速减速直到静止。



森林冰火人游戏流程图

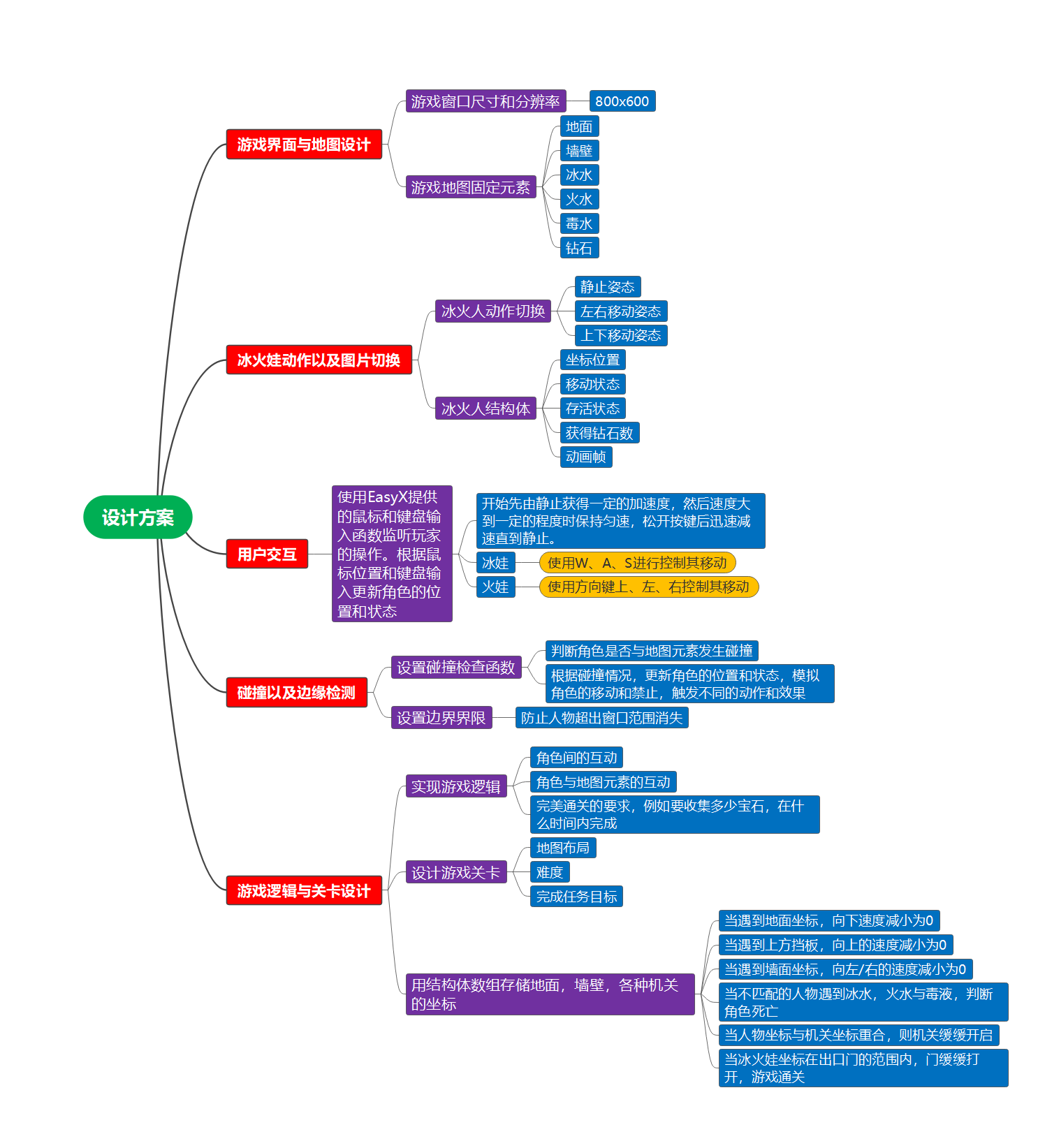
**2.5进阶功能与自主设计**

（1）进阶版地图布局与提示功能：为增加游戏的难度和趣味性，可以设计更加复杂的地图布局，加入各种谜题和障碍。同时，提供提示功能，让玩家在困难关卡中获得帮助。

（2）角色能力与道具：为冰娃和火娃添加特殊能力，例如冰娃可以冻结冰水，火娃可以点燃木块等。引入道具系统，让玩家能够获得临时性的加强能力或帮助。

（3）时间挑战与限时任务：在某些关卡中，加入时间限制，让玩家在有限的时间内完成任务。 设计限时任务，玩家需要在规定时间内完成特定的目标。 （4）故事模式与剧情设计：设计游戏的故事背景，为游戏增加更多的情感和情节。引入剧情设计，通过动画或对话展现角色之间的互动。

（5）反重力与特殊物理效果：设计反重力关卡，玩家需要应对不同的物理环境和效果。引入特殊物理效果，如悬浮、弹性等，为游戏增加趣味性。



设计方案

**3.关键算法说明**

**3.1角色移动和跳跃算法**

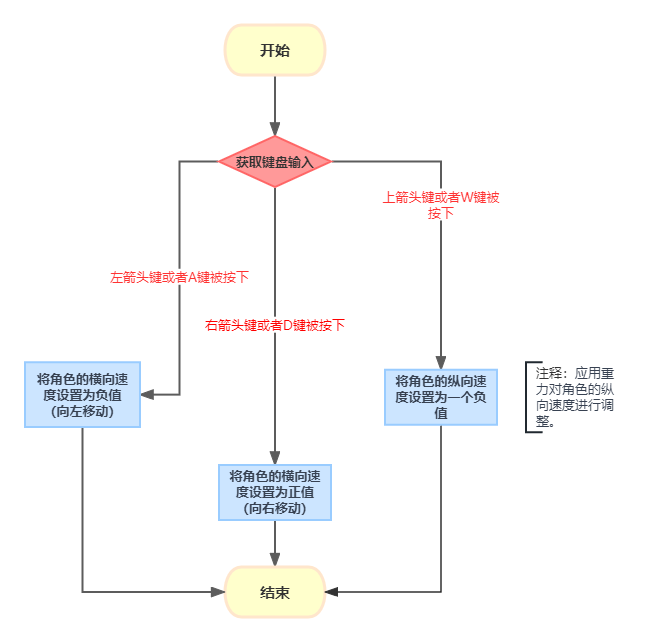
（1）设计说明：这个算法用于处理玩家控制角色的移动和跳跃行为。

a.当玩家按下左右箭头键或者A、D键时，根据键盘输入设置角色的横向速度。

b.当玩家按下上箭头键或者W键时，根据角色是否在地面上，设置角色的纵向速度，模拟跳跃动作。

c.在每一帧更新中，更新角色的位置和速度，受到重力影响，使得角色能够在地面上自由移动和跳跃。

（2）流程图



角色移动和跳跃算法流程图

**3.2碰撞检测算法**

（1）设计说明：这个算法用于检测游戏中各个对象之间的碰撞情况。

a.对于每个移动的对象，计算其边界框（矩形）。

b.对于其他的固定或移动的对象，计算其边界框。

c.使用边界框之间的相交检测，判断是否发生碰撞。

（2）流程描述

a.对于每个移动的对象（如角色），计算其边界框（矩形），包括左上角和右下角的坐标。

b.对于其他的固定或移动的对象，计算其边界框。

c.对于每一对对象的边界框，检查它们是否有交集，即是否相交。

d.如果两个边界框相交，说明两个对象发生了碰撞，需要执行相应的碰撞处理逻辑。

**4.原型设计**

**4.1主菜单原型图**

（1）描述

这个原型图展示了游戏的主菜单界面，通常包括开始游戏、选项、设置等选项。

（2）说明

a.PLAY按钮：点击这个按钮开始游戏，进入游戏的第一个关卡。

b.问号选项：点击这个按钮可以查看创作者的信息。

c.设置选项：点击这个按钮可以进入游戏设置，可以调节音量的开启与关闭。

（3）主菜单原型图



主菜单原型设计图

**4.2** **游戏界面原型图**

（1）描述

这个原型图展示了游戏的实际游戏界面，包括角色、地图、障碍物等元素。

（2）说明

a.角色：游戏的主要玩家角色，可以根据玩家的输入移动和互动。

b.地图：游戏关卡的地图，包含平台、墙壁、钻石等。

c.陷阱：游戏中的障碍物，例如火水、冰水、毒水等

d.机关道具：游戏中的道具，例如一些机关被触发后能够控制升降梯进行升降。

e.失败判定：当火娃接触到冰水，或冰娃接触到火水时，则游戏失败，当火娃或冰娃接触到有毒区域时，游戏失败。

f.胜利判定：当火娃与冰娃分别停留在相应的门（火娃为红门，冰娃为蓝门）的区域时，游戏获胜，并根据是否收集完全部的钻石获得不同的胜利结算界面。

（3）主菜单原型图



游戏界面原型设计图