

# 星露谷物语（Stardew Valley）项目简介

## 项目基本信息

建议得分权值：中高

建议团队人数：4人

游戏类型：农场模拟经营 + 角色扮演

## 游戏概述

本项目参考经典独立游戏《星露谷物语》，实现一款2D像素风格的农场模拟经营游戏。玩家将扮演一位继承爷爷农场的主角，通过种植作物、经营农场、与NPC互动等方式，体验悠闲的田园生活。

游戏融合了资源管理、时间规划、探索冒险等多种玩法，强调系统间的相互关联和玩家的策略选择，同时通过面向对象设计展现复杂的游戏逻辑。

## 基础功能（必须实现）

### 1. 角色移动与交互系统

- 角色控制
  - 支持上下左右四方向移动
  - 支持跑步/行走状态切换
  - 支持基础动画切换（站立、行走、使用工具）
- 交互机制
  - 点击/按键触发交互
  - 范围检测（与NPC、物品、建筑的交互判定）
  - 不同交互类型（对话、拾取、使用）

### 2. 农场系统

- 土地系统
  - 至少支持 30+ 可耕种格子
  - 土地状态：未开垦、已耕地、已浇水、已种植
  - 点击种植/耕地提示
- 作物系统

- 至少 **6 种作物**, 包含不同属性:
  - 生长期不同 (2天、3天、5天等)
  - 售价不同
  - 季节限制 (春季/夏季/秋季)
- 作物生长阶段显示 (种子→幼苗→成熟)
- 枯萎机制 (跨季节未收获)
- **工具系统**
  - 至少 **4 种基础工具**:
    - 锄头 (耕地)
    - 水壶 (浇水)
    - 镰刀 (收获)
    - 斧头 (砍树, 可选)
  - 工具切换 UI
  - 使用动画与音效

### 3. 时间与季节系统

- **时间流逝**
  - 游戏内时间 (每分钟代表游戏内10分钟)
  - 每日时间段: 早晨、中午、傍晚、夜晚
  - 午夜自动睡眠/保存
- **季节系统**
  - 至少支持 **2 个季节** (春季、夏季)
  - 每季 28 天
  - 季节切换时场景变化 (色调/植物)
- **日历显示**
  - 当前日期
  - 星期几
  - 季节名称

### 4. 背包与物品系统

- **背包管理**
  - 至少 **20 格背包空间**

- 物品堆叠（同类物品最多堆叠 99 个）
- 物品拖拽/整理功能

- **物品分类**

- 种子类
- 作物类
- 工具类
- 资源类（木材、石头等）

- **物品属性**

- 名称、图标
- 描述
- 售价/购买价

## 5. 商店系统

- **种子商店**

- 至少 6 种种子出售
- 按季节更新商品
- 购买界面（显示价格、库存）

- **交易系统**

- 金币货币系统
- 购买/出售判定
- 余额不足提示

## 6. 经济系统

- **资金管理**

- 初始金币：500
- 收入来源：出售作物
- 支出：购买种子、工具升级（可选）

- **出售机制**

- 出货箱系统
- 午夜结算收入
- 价格计算与显示

## 7. UI 系统

- **主菜单**
  - 新游戏
  - 继续游戏
  - 退出
- **游戏 HUD**
  - 当前金币数
  - 当前时间/日期
  - 当前季节
  - 当前体力值 (可选)
- **暂停菜单**
  - 继续游戏
  - 保存游戏
  - 返回主菜单

## 8. 存档系统

- 支持游戏进度保存
- 支持读取存档
- 保存内容：
  - 农场状态
  - 背包物品
  - 金币数量
  - 游戏时间

## 9. 音效系统

- 背景音乐 (至少 2 首：白天/夜晚)
  - 音效 (浇水、耕地、收获、购买等)
- 

## 可选功能 (自由扩展)

### 进阶功能

- **建筑系统**
  - 鸡舍/牛棚 (养殖动物)

- 谷仓（存储物品）
- 温室（全季节种植）

- **动物养殖**

- 鸡、牛等动物
- 产出鸡蛋、牛奶
- 好感度系统

- **矿洞探索**

- 多层矿洞（至少 10 层）
- 怪物战斗（基础 AI）
- 矿物采集（铜矿、铁矿）
- 战斗系统（生命值、武器）

- **钓鱼系统**

- 钓鱼小游戏
- 至少 5 种鱼类
- 钓鱼技能等级

## 社交系统

- **NPC 系统**
  - 至少 3 个可互动 NPC
  - 对话系统（多轮对话）
  - 好感度系统
  - 送礼机制
- **事件系统**
  - 节日活动（春季舞会、秋季庆典）
  - 随机事件

## 技能与升级

- **技能系统**
  - 农业技能（增加作物品质）
  - 采矿技能（增加矿物获取）
  - 钓鱼技能（增加鱼获几率）
  - 经验值与等级提升

## 高级功能

- **作物品质系统**

- 普通、优质、极品三档
- 影响售价

- **天气系统**

- 晴天、雨天、雪天
- 影响作物生长

- **成就系统**

- 追踪玩家进度
  - 解锁奖励
- 

## C++ 特性应用规划

为满足课程要求，项目将重点应用以下 C++ 特性：

### 必须使用（至少 3 条）

#### 1. STL 容器

- `std::vector<>` 存储作物列表、背包物品
- `std::map<>` 存储物品 ID 与数据的映射
- `std::queue<>` 实现事件队列

#### 2. 类与多态

- `Item` 基类 → `Seed`、`Crop`、`Tool` 派生类
- `GameObject` 基类 → `Player`、`NPC`、`Building` 派生类
- 虚函数实现不同对象的 `update()` 和 `render()`

#### 3. C++11 特性

- `auto` 关键字简化迭代器使用
- `lambda` 表达式用于回调函数
- `智能指针` (`std::shared_ptr`、`std::unique_ptr`) 管理对象生命周期
- `enum class` 定义游戏状态、季节类型

#### 4. 模板（可选加分）

- 通用的资源管理类 `ResourceManager<T>`

- 对象池模板 `ObjectPool<T>`

## 5. 操作符重载（可选加分）

- 重载 `+`、`-` 实现坐标运算
- 重载 `<<` 实现自定义输出

## 6. 异常处理（可选加分）

- 文件读写异常捕获
- 资源加载失败处理

---

# 架构设计要点

## 系统模块划分

```
GameManager (游戏管理器)
├── SceneManager (场景管理)
├── TimeManager (时间管理)
├── SaveManager (存档管理)
├── AudioManager (音频管理)
└── UIManager (UI管理)
```

```
GameScene (游戏场景)
├── Player (玩家)
├── FarmManager (农场管理)
|   ├── TileMap (地块系统)
|   └── CropManager (作物管理)
├── InventoryManager (背包管理)
├── ShopManager (商店管理)
└── NPCManager (NPC管理, 可选)
```

## 核心类设计示例

```
cpp
```

```
// 物品基类
class Item {
protected:
    int id;
    std::string name;
    int price;
public:
    virtual void use() = 0; // 纯虚函数
    virtual ~Item() = default;
};
```

```
// 作物类
class Crop : public Item {
private:
    int growthStage;
    int daysToGrow;
    Season season;
public:
    void grow();
    bool isHarvestable();
};
```

```
// 单例模式 - 游戏管理器
class GameManager {
private:
    static GameManager* instance;
    GameManager() = default;
public:
    static GameManager* getInstance();
    // ...
};
```

## 技术难点预估

### 1. 地块系统的实现

- 需要二维数组管理每个地块状态
- 鼠标点击坐标转换为地块索引

### 2. 时间系统与多系统联动

- 时间流逝触发作物生长、季节切换
- 需要观察者模式或事件系统

### 3. 存档系统

- 序列化/反序列化复杂数据结构
- JSON 或二进制格式选择

### 4. AI 设计 (可选)

- NPC 行为状态机
  - 矿洞怪物的追击逻辑
- 

## 开发建议

### 分工建议 (4人团队)

- 成员A：核心框架 + 角色系统 + 地块系统
- 成员B：作物系统 + 时间系统 + 经济系统
- 成员C：UI系统 + 背包系统 + 商店系统
- 成员D：存档系统 + 音效系统 + 可选功能

## 开发阶段

### 第一阶段 (2周)：核心框架搭建

- Cocos2d-x 项目初始化
- 基础场景、角色移动
- 地块系统雏形

### 第二阶段 (1.5周)：核心玩法实现

- 作物种植、生长、收获
- 时间系统
- 背包与商店

### 第三阶段 (1周)：系统完善

- 存档系统
- UI 优化
- 音效添加

### 第四阶段 (0.5周)：测试与优化

- Bug 修复

- 性能优化
  - 文档编写
- 

## 预期成果

完成本项目后，将实现一款：

- 可完整运行的 2D 农场模拟游戏
  - 包含完整的种植→收获→出售经济循环
  - 具备时间与季节系统
  - 支持存档读档功能
  - 代码结构清晰，符合面向对象设计原则
  - 合理应用多种 C++ 特性
  - 符合 Google C++ Style 代码规范
- 

## 参考资源

- Cocos2d-x 官方文档：<https://docs.cocos.com/cocos2d-x/manual/zh/>
  - 星露谷物语 Wiki：<https://stardewvalleywiki.com/>
  - 像素艺术资源：OpenGameArt、itch.io
  - Git 协作：GitHub Flow 工作流
- 

**项目难度评估：** 中高

**预计工作量：** 约 120-160 人时

**推荐指数：** ★★★★★

该选题既能充分展示 C++ 面向对象特性，又具有良好的可扩展性和趣味性，适合作为课程项目完成。