

# 2020 高级程序设计2 期末项目说明

---

## 2020 高级程序设计2 期末项目说明

基本要求

重要说明

评分标准

选择参考选题进行开发的项目

自拟内容进行开发的项目

参考选题

开心消消乐

元气骑士

加分项

版本控制与团队协作

代码质量与安全

功能和架构

其他

助教和联系方式

附录

钦定的 git 托管网站

必须使用的 C++ 特性

必须达成的代码规范

参考资料

关于Cocos2d-x

Git 教程

Git 图形化客户端

## 基本要求

---

1. 在给定的几个参考选题中选择一个完成，或自行拟定内容，但必须是游戏类型，且使用[Cocos2d-x](#)这一C++游戏引擎库。对于自行拟定的题目，需在 **5 月 15 日**前联系助教以评估难度、工作量和可行性（联系方式见"助教和联系方式"）
2. 以个人或团队身份完成，可以在软件工程荣誉班内跨班组队。若从参考选题里选题，则团队人数限制见选题的说明；若自选题目，则团队人数不超过 4 人
3. 必须使用 git 进行版本控制，且公开发布指定的 git 托管网站之一（见"附录"），并请大家遵循Git最佳实践，保留每个人的提交记录，该记录将作为个人工作量的评判依据。
4. 实现了基本功能、可编译为可执行文件、可正常运行不崩溃
5. 不得抄袭任何代码，包括开源项目中的代码，但允许使用开源库；若以源码方式引用开源库，则开源项目的源码或头文件需放入和你编写的源代码不一样的目录中
6. 要求使用[Cocos2d-x](#)作为游戏开发框架，不得只包含控制台界面；本项目偏向于考察和锻炼代码和架构设计，请勿将重点错误地放在了精美的界面上
7. 必须贯彻且合理地使用了指定 3 条以上的 C++ 功能（见"附录"）
8. 必须达成所有指定的代码规范要求（见"附录"）
9. 需在 **5 月 17 日**前将分组信息（包含学号和姓名）、组长（仅限团队项目）、选题（或自拟课题的

- 详细描述)、项目 git 地址(可以先创建一个只有ReadMe的空项目)发送给助教(两位助教都需要发送,联系方式见"助教和联系方式")
10. 期末需进行答辩(答辩需要包括的内容、答辩时间另行通知)
11. 期末需递交项目源码、可执行程序 and 文档(文档需涵盖的内容另行公布)

## 重要说明

1. 本次项目考察同学们对于面向对象程序设计的能力,重点在于对类、继承、多态等面向对象程序设计特点的理解以及对于复杂C++类库的使用。
2. 本次项目主题为游戏,限于时间,我们对游戏的界面精美程度、游戏难度、平衡性等不做硬性要求,但应当具备基本的界面及合理的游戏难度
3. 关于代码抄袭问题,我们允许同学们借鉴开源项目中的部分代码,但应在学习他人写法的基础上,自己完成程序编写,且应在文档中注明代码的参考链接;
4. 仍然是关于代码抄袭,我们将项目进行查重,重点在于对类设计、游戏逻辑代码的查重,要求代码重复率不超过30%。一旦发现代码抄袭问题,将按照作弊处理。

## 评分标准

以团队为单位进行评分,同一个团队下的成员一般会获得比较相近的期末项目分数,但贡献非常少的团队成员可能被扣除不超过 40% 的期末项目分数。贡献程度由 git 递交历史评定。

## 选择参考选题进行开发的项目

评分项	权重
基础功能的实现程度 [*]	40% × 得分权值
基础功能以外新开发的功能的创新性	5%
基础功能以外新开发的功能的技术难度	15%
基础功能以外新开发的功能的工作量	10%
C++ 功能和特性的使用程度	15%
答辩	10%
文档	5%
以少于建议人数完成项目[**]	不超过额外15%
加分项	不超过额外 30%

[\*] 选题的得分权值(本文档给出,见"参考选题")将影响你在该项的最终得分。例如在得分权值为 0.6 的情况下,你在该项中的最终得分不会超过  $40\% \times 0.6 = 24\%$ ,因此你的期末项目总分至多为 84分(100 分计)。得分权值大于 1 时,多出来的分数作为额外加分项(不与评分项中的加分项冲突)。

[\*\*] 每个参考选题将会给出建议的团队人数,当实际团队人数少于建议人数时,每少1人,项目将获得额外5%的加分,最多不超过15%。

# 自拟内容进行开发的项目

评分项	权重
所完成内容的创新性	10%
所完成内容的技术难度	30%
所完成内容的工作量	30%
C++ 功能和特性的使用程度	15%
答辩	10%
文档	5%
以少于建议人数完成项目[*]	不超过额外15%
加分项	不超过额外 30%

[\*] 助教将会根据自拟内容给出建议的团队人数，当实际团队人数少于建议人数时，每少1人，项目将获得额外5%的加分，最多不超过15%。

## 参考选题

每个选题都给定了一个得分权值，代表项目的难度和工作量，也决定了你期末项目分数的上限。得分权值越高，完成该项目的难度和所需的预计工作量越大、完成相同数量的基础功能情况下所得的分数也会越多。请根据自己的水平选择合适的项目，避免由于选择了高难度项目毫无进展而得零分，或由于选择了太低难度项目无法得到全部分数。

在这里我们仅给出大致功能需求，具体的游戏细节请大家自行设计。对于所给出的基础功能或可选功能要求，可以不完全按照这里的描述：对于基础功能应实现大致类似的功能，对于可选功能我们仅提供部分参考；欢迎大家充分发挥主观能动性，自行设计更多有趣的功能或机制。

## 开心消消乐

得分权值：0.8

团队人数限制：<= 2人

建议人数：2人

描述：实现一个类似于[开心消消乐](#)的游戏

需要达成的基础功能：

- 游戏开始界面、背景音乐、音量控制、暂停及恢复功能、退出功能
- 至少4种不同类别的方块
- 同类方块三消及三个以上的消除，超过三个相同方块消除时产生特殊方块，特殊方块在消除时产生强力效果，比如消除一行（列）、消除周边一定范围方块、消除所有同类方块等
- 随机产生新的方块
- 方块的移动动画

- 得分统计
- 关卡步数限制
- 至少2个关卡，每次进入关卡时方块的分布是随机的

基础功能以外的可选功能（欢迎自创）：

- 本地排行榜，如得分排行榜、通关速度排行榜
- 剩余步数奖励机制，如当玩家通关时仍有剩余步数，可以在终局方块中随机产生特殊方块并消除，使用户获得更高分
- 更多类型的方块，如需要消除多次的方块、只会移动不会被消除的物品、冰块等
- 2种及以上游戏通关条件，如消除指定多个特定种类方块、消除指定数量的冰块、收集指定数量的物品等
- 奖励机制，通过关卡可以获得金币等奖励
- 特殊道具，如消除指定方块、交换两个方块等，可以通过购买或抽奖等方式获得

## 元气骑士

得分权值：1.2

团队人数限制：<= 4人

建议团队人数：3 - 4 人

描述：实现一个类似于[元气骑士](#)的Roguelike类游戏

需要达成的基础功能：

- 游戏开始界面、背景音乐、音量控制、暂停及恢复功能、退出功能
- 键盘控制
- 至少2类武器，包括不限于枪械类武器、近战武器；每类武器至少包括2种具体的武器，如枪械类武器包括不限于一次发射单发子弹的枪械、一次发射多发子弹枪械等，近战武器包括不限于可快速攻击的小范围武器、攻击速度较慢的大范围武器等
- 至少2种普通敌人类型，包括不限于近战型敌人和远程型敌人；每类普通敌人至少包括2类具体的敌人
- 至少两种道具，包括不限于恢复生命、增加护甲的道具，可以通过击败敌人获取
- 宝箱机制，开启可获得新武器或者道具等
- 安全地图与探险地图，安全地图指进入关卡前的地图，可以查看武器信息、敌人信息等，探险地图指关卡地图
- 关卡中主角死亡，探险终止并返回安全地图
- 至少包括2个关卡

基础功能以外的可选功能（欢迎自创）：

- 每次进入即使相同关卡也会有不同的游戏地图、敌人、道具等（Roguelike!）
- 游戏小地图，可以显示已经探索过的地图，可以参考元气骑士的小地图
- 多个可选主角，每个主角拥有不同的属性加成（近战攻击速度提升、枪械子弹速度提升、生命值增加等）或者主动技能（移速增加、翻滚躲子弹）
- buff机制，主角可以获得多种buff，包括不限于移速增加、无敌等
- debuff机制，主角会获得多种debuff，包括不限于移速减慢等
- 金币机制，击败敌人可以获取金币

- 主角等级系统，高等级主角拥有更强的属性或能力，主角可以通过消耗金币升级（在安全地图中）
- 安全地图中商店系统，可以在进入关卡前购买或抽取特殊道具、武器等
- 探险地图中的老爷爷系统，可以在关卡中购买/获赠礼物，如武器/道具/buff等
- 特殊地形，包括不限于可被破坏区域（如木墙）、移速受限区域（如沼泽）、通过受伤区域（如岩浆）、机关地形（如时不时出现的地刺）等
- BOSS型敌人
- 秘籍，如十倍攻击力、无敌等

## 加分项

---

加分项可叠加；请在答辩前告知助教核查项目的加分项。

## 版本控制与团队协作

- 符合一些 git 最佳实践（如 commit 历史干净规范、commit 描述规范、commit 包含且仅包含一项功能、使用 Pull Request 功能或分支进行团队协作等）；
- 团队成员分工平等、合理；

## 代码质量与安全

- 对于复杂逻辑使用单元测试验证正确性；
- 使用断言验证程序性质；
- 合理的异常抛出与处理

## 功能和架构

- 界面精致；
- ==Super Bonus: 能移植或直接运行在 Android 或 iOS 平台并参加竞赛；==
- 项目目录结构良好、清晰；

## 其他

- 没有内存泄漏；
- （助教自行通过界面玩耍程序时）程序很少崩溃；
- 贯彻且合理地使用了下列 3 条以上的 C++11 或更高的 C++ 新特性：
  - 初始化列表
  - 类型推断 (`auto` / `decltype`)
  - 基于范围的 `for` 循环
  - 智能指针
  - 常量表达式 (`constexpr`)
  - Lambda 表达式
  - 右值引用
  - 字符串字面量
  - 其他 C++11 或更高的新特性
- 其他你认为可以作为加分项的亮点。

## 助教和联系方式

---

- 姜逸之
  - 负责：加分项认定、项目难度评估
  - QQ：937522035
  - 邮箱：[1931566@tongji.edu.cn](mailto:1931566@tongji.edu.cn)
- 林程
  - 负责：其他事宜
  - QQ：772812969
  - 邮箱：[lincheng\\_619@163.com](mailto:lincheng_619@163.com)

## 附录

---

### 钦定的 git 托管网站

你的项目必须使用 git 进行版本控制（git 图形化客户端见"参考资料"），且公开发布以下 git 托管网站之一，否则扣除不超过总分 100% 的分数：

- GitHub（推荐）
- GitLab（推荐）
- Bitbucket
- Coding.net（若上述三个网站访问困难可考虑使用这个）

### 必须使用的 C++ 特性

你的项目必须贯彻且合理地使用了下列 3 条以上的 C++ 功能，否则扣除不超过总分 100% 的分数：

- STL 容器，如 `std::vector`，`std::map` 等；
- 迭代器；
- 类和多态；
- 模板；
- 异常；
- 函数重载；
- 操作符重载；
- C++11 或以上功能。

### 必须达成的代码规范

你的项目必须达成以下所有代码规范要求，否则扣除不超过总分 50% 的分数：

- 代码在缩进、命名等方面基本遵循了统一和一致的風格（如 Google C++ Style）；
- 正确、合理地使用了 C++ 风格类型转换（如 `static_cast`、`dynamic_cast`）且没有使用 C 风格的强制转换；
- 尽可能地使用了 `const` 和引用；
- 类的设计合理规范。

## 参考资料

---

## 关于Cocos2d-x

- [官方文档](#)
- 

## Git 教程

- [官方文档](#)
- [GitHub 和 SourceTree 入门教程](#)

## Git 图形化客户端

对于新手，推荐使用图形化界面进行 git 版本控制，例如：

- [SourceTree](#)（跨平台）
- [SmartGit](#)（跨平台）
- [GitHub Desktop](#)（跨平台）