

# 第一次课程设计

## 0. 写在前面

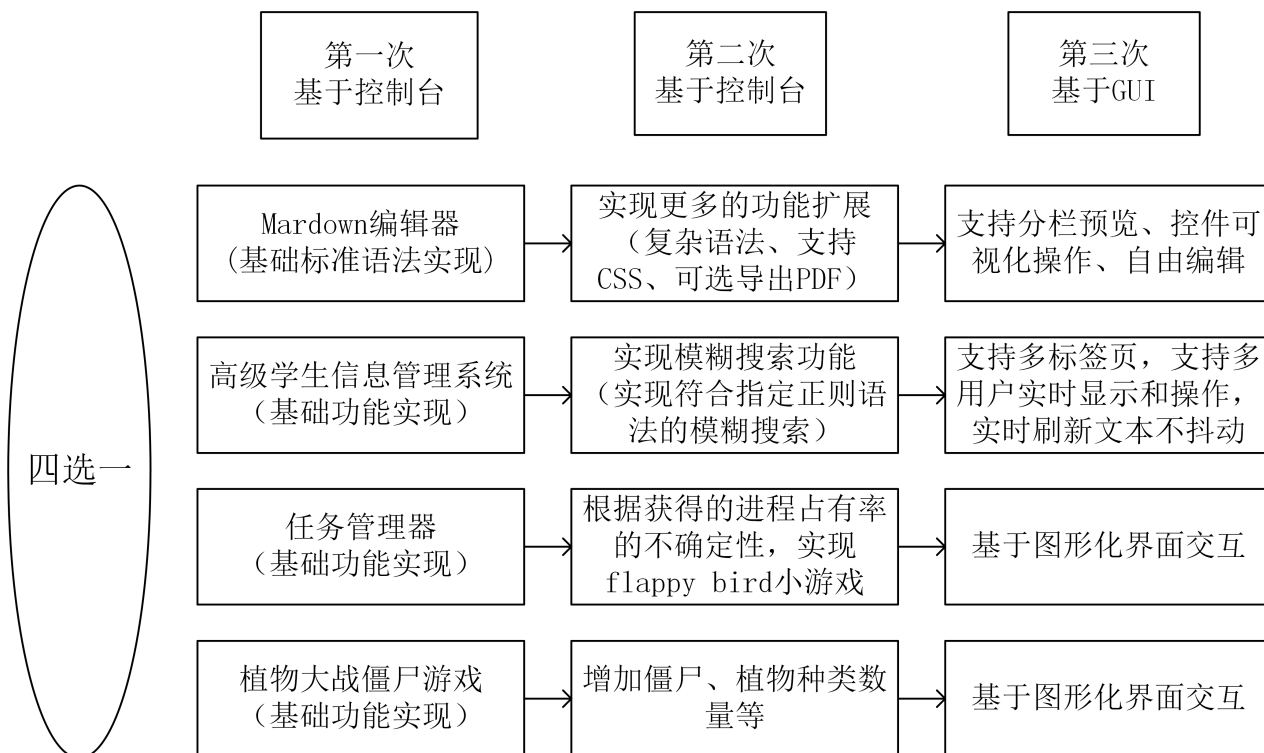
本学期“高级程序设计”课程将会计划安排三次课程设计作业，具体安排时间如下（现场验收时间待定，由各组助教安排）：

- 第一次，第4-6周 3月23日~4月13日
- 第二次，第7-10周 4月13日~5月11日
- 第三次，第11-14周 5月11日~6月8日

三次作业的内容是关联递进的，请大家根据自己的能力与喜好选择实现下面四条路线之一即可（路线大致内容如下图所示）。我们分别会在3月23日、4月13日和5月11日公布各次课设具体的题目要求。本次为第一次课设，为期3周。

如果对现有的三个课设题目不满意或者感到不合适，我们也允许大家自拟题目，但需经过助教和老师的讨论同意后，方可进行。

请注意使用面向对象的设计方法来思考设计内容中的对象、类以及它们之间的关系。



## 1.1 Markdown编辑器

Markdown是一种纯文本格式的标记语言，通过简单的标记语法，它可以使普通文本内容具有一定的格式，使编辑者摆脱格式的困扰，专注于内容的编写，并且可以导出成HTML、PDF等多种格式。现在我们需要实现一个简单的Markdown编辑器，可以在控制台录入Markdown格式的文件，对其进行解析，并生成HTML格式的输出。

本次课设要求从Markdown标准语法中选择以下语法进行实现：

### 1.1.1 标准语法子集

1. 标题：# 后 加个空格再加标题，# 个数表示标题级别，最高支持六级标题

# 一级标题

## 二级标题

### 三级标题

2. 字体设置：

斜体：对要加粗的文字左右分别用一个 \* 号包起来

加粗：对要倾斜的文字左右分别用两个 \* 号包起来

斜体加粗：对要倾斜加粗的文字左右分别用三个 \* 号包起来

*\*斜体文本\**

**\*\*加粗文本\*\***

***\*\*\*斜体加粗文本\*\*\****

3. 多种语法混用

# 一级*\*斜体标题\**

### 1.1.2 基本要求实现

1. 从控制台录入Markdown格式的文本，并保存生成 .md 文件

2. 载入生成的 .md 文件，根据上述1.2.1中定义的Markdown语法子集进行解析，并生成HTML文件（需要保证可以通过浏览器打开，正常展示）。

## 1.2 高级学生信息管理系统

### 1.2.1 题目说明

本题目要求完成一个基于控制台的学生信息管理系统，实现对学生信息的添加、删除、查找、修改、显示、保存和读取等操作。学生的信息包括学号、姓名、院系、班级、课程和成绩等。

### 1.2.2 题目要求

1. 添加和删除：系统将提示用户输入新添加学生的信息；由用户输入要删除的学生的学号，然后调用删除函数，删除该名学生的资料
2. 查找：用户输入要查找的学生的学号，系统调用查找函数进行查找并显示结果。
3. 修改：用户输入要修改的学生的学号，然后系统用修改函数查找，显示该名学生的资料，然后系统提示用户输入需要修改的项目和新的资料。
4. 显示：用户输入要查看的院系或班级或某课程，系统调用查找函数查找并显示结果。
5. 保存：将所有的学生信息保存到一个文件当中。
6. 读取：从文件中读取学生信息并录入到系统中。

### 1.2.3 题目扩展（不作要求）

1. 实现对某一门课程的平均成绩、选修人数等统计信息的查询和显示。
2. 提供对查询结果导出到文件的支持。

## 1.3 任务管理器

### 1.3.1 题目说明

本题目要求完成一个基于控制台的进程管理器，可以参考Windows进程管理器，实现进程信息的记录、显示、查找、排序等操作。进程的信息包括进程ID（PID）、进程名字、进程位数（32位程序/64位程序）、进程的CPU占有率计算等。

### 1.3.2 题目要求

1. 记录：将获取到的进程信息每1分钟保存到某个文档中，文档名称为当前的时间戳。
2. 查找：用户输入要查找的进程ID，系统调用查找函数进行查找并显示结果。
3. 显示：当用户输入指令时，或者每一分钟，显示当前的最新进程信息，包括进程的ID、进程的名称、进程的位数以及进程的CPU占有率。
4. 排序：可按照进程PID大小、进程名字、CPU占有率进行排序。

### 1.3.3 参考提示

1. 进程CPU占有率可以通过记录一段时间间隔内，进程使用CPU的时间来计算。
2. 可能需要使用到的系统API如下，网上有很多相关的信息可以参考，此外CPU占用率的实现方法较为多样化，鼓励同学们可以尽可能使用不同的方法来实现这一点，计算结果越接近Windows任务管理器越好。

**GetSystemInfo**          主要用它来获取系统中CPU核心个数

**OpenProcess**          用来打开指定进程的句柄

**GetProcessTimes**      根据OpenProcess返回的句柄，获取进程的KernelTime和UserTime（可用于进程的CPU占用率计算）

**GetSystemTimes**      获取总的CPU时间IdleTime、KernelTime和UserTime，可用于系统总的CPU占用率计算（注：多核CPU中返回的是所有CPU核时间的总和）

**NtQuerySystemInformation**      这是个native api，可以获取到许多信息

## 1.4 植物大战僵尸

塔防游戏植物大战僵尸曾一度风靡全球，现在让我们仿制一个简单的版本吧！

对于玩过这个游戏的玩家来说，你只需要知道模仿设计的是：

- 前院场景（纯草地无水池）
- 白天（系统会产生自然光）
- 无尽模式（需要记分牌）对于没有玩过这个游戏的玩家，建议先去4399体验一下，但这里还是简要说明一下游戏逻辑：
- 按照一定的策略随机产生僵尸，从马路进入玩家的庭院，吃掉玩家种植的植物，以庭院左边的底线为目标前进
- 玩家通过收集阳光、种植植物反击以攻击消灭僵尸并保护房子
- 游戏失败：任何一只僵尸进入了庭院左边的底线
- 游戏胜利：由于是无尽模式所有没有胜利条件，目标是能够持续抵挡僵尸的进攻，已获得更多的累计积分

### 1.4.1 植物设计参考

自行设计购买各种植物所花费的阳光数、植物的生命值和攻击力以及冷却时间等属性。可以自行设计植物的种类和效果，不限于以下所列出的。

- 攻击型
  - 豌豆射手：对僵尸发射豌豆造成伤害
  - .....
- 防御型
  - 坚果：生命值高，阻挡僵尸前进一段时间
  - .....

- 炸弹型
  - 窝瓜：当有僵尸出现在地块内之后，炸毁所在地块的所有僵尸
  - .....
- 效果型
  - 向日葵：每隔一定的时间产生一定数量的阳光

#### 1.4.2 僵尸设计参考

自行设计僵尸的生命值、攻击力、速度等属性。可以自行设计僵尸的种类和效果，不限于以下所列出的。

- 普通僵尸
- 路障僵尸：有较高的防御力
- 撑杆僵尸：可以跳过遇到的第一个植物
- .....

#### 1.4.3 商店设计参考/用户交互



系统产生的自然光和向日葵产生的阳光都是自动收取，玩家唯一需要做的就是购买植物和种植植物。

植物购买、地块选择和植物种植的参考流程如下：

1. 捕捉键盘字母b表示购买某种植物
2. 焦点移动至庭院，通过上下左右键选择空的地块
3. 捕捉回车键确认种植，字母x取消

#### 1.4.4 界面设计参考

庭院可以设计成如下所示的3行7列的布局，下面再加一行商店与记分牌。界面仅供参考，可自由设计。

```
#####
#
#           #           #           #           #           #           #
# 向日葵  # 向日葵 #           # 豌豆射手 #  # 僵尸 # 路障僵尸
#           #           #           #           #           #           #
#####
#           #           #           #           #           #           #
# 向日葵  #           #           # 豌豆射手 #           #  # 僵尸 #
#           #           #           #           #           #           #
#####
#           #           #           #           #           #           #
```

```
# 豌豆射手# 豌豆射手 #          #          #          # 路障僵尸# 僵尸 # 僵尸
#          #          #          #          #          #          #          #
#####
```

=====

[商店] 阳光总数： 225 | [得分]

-----

向日葵50 豌豆射手100 | 75

=====

### 1.4.5 基本实现要求

在本次设计中，只要求实现最基本的要求：

- 庭院布局至少3行7列
- 实现每隔一定时间，系统会产生自然光
- 实现一种僵尸：普通僵尸
- 实现僵尸的三种属性：生命值、攻击力、速度
- 假定每个地块中只能有一只僵尸
- 实现一种植物：豌豆射手
- 实现植物的三种属性：购买所费阳光数、生命值、攻击力
- 实现植物的购买、地块的选择和植物的种植
- 实现当前局的记分牌

## 2. 说明

- 本次课程设计是**基于控制台**的应用，须采用**面向对象**的方法实现，但对继承、多态、泛型等高级特性的使用**不作要求**。
- 需提交课程设计报告（PDF文档，不要提交PPT），课程设计报告尽量详细，主要描述课程设计的设计实现过程，包括但不限于以下方面：
  - 课程设计的主要内容、目标和设计思路
  - 主要类的设计，包括类的数据与操作，以及类之间的关系
  - 程序的功能亮点与运行操作方法
  - 遇到的问题与解决方案
- 课程设计的评分主要参考以下方面：
  - 课程设计报告的书写
  - 程序功能的实现与现场演示
  - 程序的总体框架设计

### 3. 提交方式

- 提交内容包括：
  - 完整工程项目目录，包括所有的代码文件、头文件等
  - 程序的可执行文件（包含依赖文件），可直接运行
  - 课程设计报告
- 将上述内容一起打包成压缩文件并命名为“学号\_姓名\_高程课设一.zip”提交

提交截止时间：2021年4月13日 23 : 59