

第二次课程设计

0. 写在前面

本次为第二次课设，为期四周，具体时间为 第7-10周 4月13日~5月11日。

要求是基于控制台的程序，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，并合理运用**继承**、**多态**和**泛型**等特性。

1. 具体题目要求

1.1 Markdown编辑器

在第一次课设的基础之上，要求实现如下扩展。

1.1.1 标准语法子集扩展

1. 超链接：

[系主页](https://cs.nju.edu.cn)

2. 列表：

- 无序列表：在列表项文本前添加 + 或 - 或 * 和空格即可
- 有序列表：在列表项文本前添加数字和空格即可
- 列表之间多级嵌套

+ 列表内容

- 列表内容

* 列表内容

1.列表内容

2.列表内容

3.列表内容

* 一级无序列表内容

1.二级有序列表内容

2.二级有序列表内容

3. 引用：在被引用的内容前添加 > 即可

> 引用内容

4. 分割线：三个或者三个以上的 * 或者 - 即可

5. 表格：语法示例如下，第二行分割表头和内容，其中左边加 : 表示文字居左，两边加 : 表示

文字居中，右边加：表示文字居右

| 表头 | 表头 | 表头 |

| :- | :-: | -: |

| 内容 | 内容 | 内容 |

6. 多种语法混用，例如：

+ 列表****斜体加粗内容****

1.1.2 CSS样式扩展

- 可设置字体、颜色等
- 超链接样式：选中和悬停链接时的背景颜色，未访问链接和访问过链接的不同样式等

1.2 高级学生信息管理系统

在第一次课设的基础之上，要求实现如下扩展。

1.2.1 模糊搜索

1. 择一匹配

"|"：选择符号左右两边的其中一项，例如“张三（一|二）”将匹配到“张三一”或者“张三二”。

2. 单字符匹配

"."：匹配对应位置的任意单个字符（换行符除外），例如“张.三”，将匹配到“张三一”，“张三二”，“张三三”，“张四三”，.....

3. 首尾匹配

^ 接字符串，表示以该字符串开始，例如“^张”，将匹配到“张三一”，“张三二”，“张三三”，..... “^MF19”将匹配到学号开头为MF19的学生

\$ 接字符串，表示以该字符串结尾，例如“\$三”，将匹配到“王三”，“张三”，“李三”，.....

4. 多种规则混合

对上述规则进行组合，例如“^夏..”可以匹配到姓“夏”的名字为三个字的同学

1.2.2 高级搜索

可设置多种组合的筛选规则

&& 同时满足：学号="^MF19" &&姓名="张.."

|| 满足其一：选修课程="高级程序设计" || "基础程序设计"

!= 不等于：学号!="^MF19"

1.3 任务管理器

任务管理器中实时采集的各个进程的CPU利用率可以认为是一种随机数，随机数在小游戏中是非常重要的一项参数，例如Flappy Bird。在Flappy Bird游戏中，会出现不同长度的管道，当控制按键时，小鸟会上升，当没有操作时，小鸟会下降，需要控制小鸟不撞管道和地面。

- 管道分为上下管道，中间有空隙，其中空隙占30%。
- 小鸟的外观设计不必过于复杂，可以用某个字符代替，或者用矩形。
- 需要记分牌，每次躲过管道或者避开其它障碍物，可以得分。
- 按下某个键盘，可以控制小鸟上升，按一个按键上升一格，同时按两个按键上升两格。

1.3.1管道参考设计

- 视野内应该至少出现三列管道。
- 下管道高度为进程CPU使用率（将所有进程的CPU使用率降序排列，选前几个指定数量的进程CPU使用率）。
 - 为了避免CPU使用率过小，影响游戏的可玩性，可以设计数值放大机制，例如CPU利用率在0%~10%的可在该数值的基础上乘以20，最终数值应控制在总高度的70%以内。
 - 如果系统中进程的CPU利用率均为0，可自行编写辅助小程序，例如定时运行循环的进程，可帮助提升CPU利用率。
- 中间空隙，应占总高度的30%，推荐总高度应大于等于10行，使得中间空隙足以让小鸟通行。
- 上管道的高度应为总高度的70%减去下管道的高度。

1.3.2 小鸟参考设计

- 小鸟的初始高度应为总高度的中间。
- 小鸟的控制，为了方便，统一规定，当按下键盘“W”键时，小鸟可以上升一格，同时按下“W”，“E”键时，可以上升两格。
- 小鸟的外观可以用矩形，或者可显示的字符代替。
- 小鸟的积分，每增加5分，可以发射子弹，子弹可以打掉一格管道。统一规定按下“L”键发射子弹。

1.3.3 计分板参考设计

- 计分板应放在显眼的地方，每次通过一列管道，分数加一分。
- 分数每增加5分，子弹数量将加一，子弹数量在计分板上标识，例如 “积分：10（=> 2）” 括号中的数字为子弹数量。

1.3.4 参考界面设计

#####

```

    ||          ||          ||
    ||          ||          ||
    ||          ||          ||
    ||
    >>          =>  ||
                    ||          || | |
    ||          ||          ||
    ||          ||          ||
```

|| || ||

#####

=====

总分： 225 (=> 15)

=====

1.4 植物大战僵尸

在第一次课设的基础之上，要求实现如下扩展。

1.4.1 植物设计扩展

- 攻击型
 - 双发射手：发射速度是豌豆射手的两倍
 - 寒冰射手：造成伤害的同时减速僵尸的移动
- 防御型
 - 坚果墙：生命值高，阻挡僵尸并承受伤害
 - 高坚果：生命值是坚果墙的两倍，并且能阻止撑杆僵尸跃过
- 炸弹型
 - 窝瓜：当有僵尸出现在地块内之后，炸毁该地块的所有僵尸
 - 樱桃炸弹：炸掉周围3 x 3地块上的所有僵尸
- 效果型
 - 大蒜：啃食过大蒜的僵尸会被驱赶到邻近行
 - 南瓜头：种在其他植物上，以免被僵尸吃掉

1.4.2 僵尸设计扩展

- 路障僵尸：有较高的防御力
- 读报僵尸：报纸被打掉后，移动速度加快
- 撑杆僵尸：可以跳过遇到的第一个植物
- 小丑僵尸：以一定概率自爆，并炸毁周围3 x 3地块上的植物
- 投石僵尸：用篮球远程攻击植物，待篮球投完后会缓慢开车碾压植物

1.4.3 商店设计扩展

- 商店中植物的购买需要有冷却时间
- 玩家可用键盘控制在商店内进行上下左右的选择

1.4.4 界面设计扩展

- 显示植物或者僵尸的剩余生命值
- 每个地块中可以有多多个僵尸

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_高程课设二.zip”提交

提交截止时间：2021年5月11日 23 : 59