《面向对象程序设计实践（C++）》综合实验

**单词消除游戏系统设计与开发**

**单词消除游戏总体介绍**

单词消除游戏由两类参与者组成：闯关者（即游戏玩家），出题者（为游戏增加游戏中使用单词）。游戏规则为，游戏每一轮，程序会根据该关卡难度，显示一个单词，一定时间后单词消失。闯关者需要在相应地方输入刚刚显示并消失的单词，如果闯关者输入正确（即闯关者输入的单词与刚刚显示的单词完全一致，包含大小写）则为通过。一关可以由一轮或者多轮组成。

**题目一 （20分）**

闯关者属性要求必须含有：闯关者姓名、已闯关关卡数、闯关者经验值、闯关者等级，若有需要可以自行添加其他属性。

出题者属性要求必须含有：出题者姓名、出题者出题数目、等级，若有需要可以自行添加其余属性。

具体功能包括：

* 实现闯关者，出题者本地的注册、登录。
* 程序支持多人注册，同一时间只有一人可以登录。
* 实现游戏规则：出题者增加游戏中使用单词。游戏每一关，程序会根据该关卡难度，显示一个单词，一定时间后单词消失。闯关者需要在相应地方输入刚刚显示并消失的单词，如果闯关者输入正确则为通过。
* 任何角色均可查询所有闯关者、出题者，按照属性查找相应闯关者、出题者。
* 可以根据闯关者闯过关卡数、经验、等级等对闯关者排名，根据出题者出题数目、等级对出题者排名。

要求：

* 采用面向对象的方式，使用类设计。
* 在设计类时请注意类的继承关系，关注闯关者、出题者的关联（闯关者与出题者有共同的基类）。
* 使用文件或者数据库作为存储对象，自行设计文件格式，或者数据库表结构。
* 在题目的要求下自行扩展功能，功能扩展适当者酌情加分。

**题目二（20分）**

闯关者即为游戏玩家，已经注册并登录的玩家可以在系统进行单词消除游戏。每一关的难度要有所增加，体现为如下三个条件中的一个或者多个：1、单词难度可以递增或者持平（即长度加长或不变）； 2、进行轮数增多（即单词数目增加，如：前三关仅仅通过一个单词就过关，后续需要通过两个、三个甚至更多才过关）； 3、单词显示时间缩短（随着关卡的增加显示时间越来越短）。

闯关者每闯过一关，增加一定经验值。经验值会根据闯过的该关卡的关卡号、该关的闯关耗费时间共同决定。当经验值累计到一定程度闯关者等级增加。闯关失败需要重新闯该关。

游戏自带词库，而且已经注册的出题者可以为系统出题，即增加词库的新词，已经存在的单词不能再次添加（词库中的单词构成一个单词池，但建议根据单词的长度来组织存储。每次出题时，系统从该单词池中按照关卡难度随机的选择相应长度的单词）。每成功出题一次，更新该出题者的出题数目。出题者等级根据出题人成功出题数目来升级。

* 必须在题目一的基础上进行修改。
* 请根据要求设计每一关的出题方式，注意随着关卡数增加，题目难度增加。请合理处理出题人新添加新词的使用方式，并且新加词组不会影响游戏难度。
* 设计**闯关者经验值，等级增加策略。设计出题者等级升级策略**。

**题目三（10分）**

在题目二的基础上，将游戏由本地单机，扩展为服务器多人游戏平台，使用客户端/服务器的方式，同一时间可以多人登录系统。将所有闯关者、出题者信息保存在服务器。

要求：

* 必须在题目二基础上进行修改。
* 使用socket进行通信。
* 需要完成服务器端程序，以及客户端程序。客户端可以启动多个同时与服务器交互，要求服务器具有并发处理能力。

**学生可以自行扩展其他功能，扩展功能可根据合理性和实现情况酌情加分。**

加分功能举例：

* 除单人游戏外，增加双人对战游戏，要求参与闯关者均已经登录，双人同时面对一个单词，最先打出正确单词者获得经验增长，在双人对战中获胜所获得的经验增长 要高于 从同等难度的单人游戏中所获得的经验增长，失败者则需要扣除一定经验值。
* 可以查看同时在线的游戏闯关者，可以挑战在线的游戏玩家，被挑战者接受挑战后进入双人对战。