|  |  |
| --- | --- |
|  |  |
| **学 号：** | 0121801100913 |

****

**软件工程基础实验**

|  |  |
| --- | --- |
| **学 院** | 计算机学院 |
| **专 业** | 软件工程 |
| **班 级** | 软件zy1802 |
| **姓 名** | 张淳慷 |
| **指导教师** | 袁胜琼 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 2020 | 年 | 7 | 月 | 6 | 日 |

实验课程名称：\_\_软件工程基础实验\_\_\_\_

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验项目名称** | | **结对编程** | | | **实验成绩** |  |
| **实 验 者** | | **张淳慷** | **专业班级** | **软件zy1802** | **组 别** |  |
| **同 组 者** | | **刘怡博** | | | **实验日期** |  |
| 第一部分：实验预习报告（包括实验目的、意义，实验基本原理与方法，主要仪器设备及耗材，实验方案与技术路线等）  **一、实验目的**  （1）体验敏捷开发中的两人合作。  （2）进一步提高个人编程技巧与实践。  **二、实验内容及要求**  （1）选择一个程序实例，练习结对编程(pair programming)实践；  （2）要求学生两人一组，自由组合。每组使用一台计算机，二人共同编码，完成实验要求。  （3）要求在结对编程工作期间，两人的角色至少切换 4 次；  （4）编程语言不限，版本不限。建议使用 Python 或 JAVA 进行编程  **三、程序问题描述**  （1）贪吃蛇小游戏  游戏界面为一个方格图，用户可通过上下左右键控制一个长方形物体（蛇）在方格图中上下左右移动，经过任意位置随机生成的一个特殊方格（蛋）可加长长方形物体并得分，直至长方形物体碰到界面边界游戏结束。  四、主要仪器设备  PC机、eclipse | | | | | | |
| 第二部分：实验过程记录（可加页）（包括实验原始数据记录，实验现象记录，实验过程发现的问题等）  **算法设计思路：**  使用eclipse开发。因为游戏比较简单，使用一个Java包即可。有DrawView(画板类)、GameControl(游戏控制类)，GameRunThread(线程类)，Node(点类)、Snake(蛇类)、SnakeGameView(游戏面板类)。  总的来说，整个贪吃蛇建立的基础其实就是通过作出一个平面坐标图，以坐标图上的每一节点来代表其是蛋或者是蛇的基本身体结构，设置键盘监听器一判断方向键的输入时多少，调用单线程进行不断的更新，而后通过采用不同颜色的画笔对其加以修饰以得出最终的成品图。  具体的建议实现方法就是使用DrawView类重写Panel方法画面板，GameControl类添加键盘监听实现对蛇方向的控制，GameRunThread类继承Thread接口实现面板刷新，Node类定义点，Snake类定理蛇，内含蛇的方向、身体、头、吃蛋等方法，SnakeGameView类继承Frame类画出游戏面板  以下给出Snake类的几个重要思路讲解：  **Snake类：**  **Snake数据结构：**    采用链表的方式保存蛇头蛇身的x,y坐标，并且将初始化的随机数决定的坐标加入新建成的snake链表中进行蛇身的初始化，而后通过DrawView里的画笔将其画出。  **snakeMove：**    设定SnakeMove函数根据键盘监听器得到的数据以此来更改下一个节点所在的平面坐标图的位置，后利用addfirst()函数将其加入snake的数组中的第一位，并将Snake数组内的最后一位删去后通过DrawView类里采用不同的颜色画笔将背景色进行与蛇身体颜色上相应的更改，一次在视觉效果上达成蛇移动的效果。  **eatEgg：**    首先通过if判断蛇头的x，y坐标是否与蛋所在的坐标相同，如果相同即采用add方法将蛋的坐标加入snakebody的链表中而后需要调用snakeGameView类中的方法进行蛋的位置的再次随机设定，并且将jpanel面板上显示的分数进行相应的增加后完成。  **SnakeRunInterFace：**    通过if语句判断蛇身的x或者是y坐标是否有超过当初所画出坐标图的x与y的最大值，如果超过则将游戏的判断进程gameState设为False从而停止线程的运行从而使整个游戏结束。  **实验结果：**    **在实验中发现的问题：**   1. 结对编程时交流十分重要，交流好可提高编程效率，反之则会降低效率。 2. 由于第一次进行结对编程不太习惯，两个人的默契程度和对于流程的把握尚需提高。 3. 最大的问题就是彼此的编码习惯有很大的不同，所以每一次都会花费大量的时间去了解对方采用的不同的变量命名以及一些类的设计思路，有的时候与自己所想的类的设计思路不同的话，其实有的时候还会对对方所写的代码进行相应的更改一符合自己的编码习惯，一开始这样一来二去反而降低了办事的效率，但是在之后通过交流的方式定下了开发程序的基调以及一些相应变量的命名的确定最终是成功提高了开发效率，在很快的 时间之内就完成了贪吃蛇的编码。   **结对编程过程：**   1. 首先两人一起进行程序设计。 2. 两人共同复习了上半学期软件工程基础的结对编程知识，后进行结对编程，一人写代码，一人进行导航，每一小时交换位置。   **工作照片：** | | | | | | |
| 第三部分 结果与讨论（可加页）  **一、实验结果分析（包括数据处理、实验现象分析、影响因素讨论、综合分析和结论等）**  实验现象分析：  1.实验结果满足需求要求。  2.结对编程前期比较艰难，应与两人不太适应新的合作模式有关，后期熟练后效率较前期有所提高。  3.两人通过一人领航一人编码的结对编程模式，使个人任务简单化、单一化，提高编程效率  **二、实验小结及体会**  不得不说这是自己第一次尝试去真正进行结对编程，或者说应该是第一次不是说项目完完全全由自己完成而是需要与别人合作去共同完成一个项目，整体下来的感觉就是交流真的很重要，因为一开始的时候因为都是一些，怎么说呢，学习上的关系会导致说交流整个并不是很流畅，因为毕竟不是很熟，有的时候也放不开，于是做的效率就会很低，毕竟相当于不仅要自己写一遍代码，还需要是跟着别人的代码习惯和逻辑进行理解，再加上疫情的原因，本来交流也是会有障碍，不过好在之后通过不断的对于项目本身的讨论压过了彼此之间互相不熟悉的问题，讨论出了彼此应该有的工作模式，由我来进行主要的编码工作，它来进行一个基本的观察以及相应错误的点出以及更改，之后便慢慢的就解决了交流的问题，整个项目的效率也就高了起来，最终顺利的完成了这一项目。怎么说呢，实验完了最大的感受就是交流这项能力真的十分重要，在这次开发中，交流的深入与否直接决定了最终项目完成的时间以及质量的好坏，而且从这次结对编程扩展开来讲，不仅仅只是合作编程中交流的能力十分重要，其实贯穿自己整个人生历程中，学会与人交流这个能力都是至关重要的，可以十分有利的提高自己的办事效率以及方便自己其他的各种事项，所以说整体下来的体验真的还是说特别丰富的，从一个简单的结对编程感受到了之后为人处世的重要道理，还是十分令我兴奋的，同时也更期待着接下来的结对编程。  **成绩评定表：**   |  |  |  |  | | --- | --- | --- | --- | | **序号** | **评分项目** | **满分** | **实得分** | | **1** | **实验报告格式规范** | **2** |  | | **2** | **实验报告过程清晰，内容详实** | **4** |  | | **3** | **实验报告结果正确性** | **2** |  | | **4** | **实验分析与总结详尽** | **2** |  | |  | **总得分** | **10** |  | | | | | | | |

实验课程名称：\_\_软件工程基础实验\_\_\_\_

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验项目名称** | | **单元测试** | | | **实验成绩** |  |
| **实 验 者** | | **张淳慷** | **专业班级** | **软件zy1802** | **组 别** |  |
| **同 组 者** | | **刘怡博** | | | **实验日期** | **2020.5.16** |
| 第一部分：实验预习报告（包括实验目的、意义，实验基本原理与方法，主要仪器设备及耗材，实验方案与技术路线等）  **一、实验目的**  1) 掌握单元测试的方法；  2) 学习 JUnit 测试原理及框架；  3)掌握在 Eclipse 环境中加载 JUnit 及 JUnit 测试方法和过程。  **二、实验内容及要求**  1) Eclipse 中 JUnit 的使用  2)利用 JUnit 对“实验一”中的程序各功能，进行单元测试。  **三、实验基本原理和方法**  1)利用Junit模块进行测试，先对于工程添加Junit类，然后在其中添加用于测试的代码。Eclipse集成了Junit，可以非常方便的编写Test Case.而且其中自带了一个Junit插件，不用安装就可以在项目中测试相关的类，并且可以调试测试用例和被测类。  2)在贪吃蛇这个程序中需要测试的功能主要是snake类中贪吃蛇对于上下左右输入的反应以及触墙死亡和吃下点身体增长功能的测试，还有就是Node类中的取得节点函数是否可以正常运行而其中使游戏正常运行的关键就是贪吃蛇对于上下左右输入的反应，那么不妨实验时设置如下用例，包括：  上下左右输入的反应：  1.用户输入上，贪吃蛇向上移动  2用户输入下，贪吃蛇向下移动  3用户输入左，贪吃蛇向左移动  4用户输入右，贪吃蛇向右移动  触墙死亡：  1.X坐标超过限额  2.Y坐标超过限额 | | | | | | |
| 第二部分：实验过程记录（可加页）（包括实验原始数据记录，实验现象记录，实验过程发现的问题等）  四、实验流程：  首先是根据示例小组内进行Junit的学习，通过讨论不难发现：  1、测试方法上必须使用@Test进行修饰，其作用为将一个普通方法修饰为一个测试方法  2、测试方法必须使用public void进行修饰，不能带任何的参数  3、测试单元中的每个方法必须可以独立测试，测试方法间不能有任何依赖  4、有些测试在运行前需要创造几个相似的对象。在 public void 方法加@Before是因为该方法需要在 test 方法前运行。  3、如果你将外部资源在 Before 方法中分配，那么你需要在测试运行后释放他们。在 public void 方法加@After是因为该方法需要在 test 方法后运行。  **SnakeTest类：**      **NodeTest类：**    **Junit测试结果：**    第三部分 结果与讨论（可加页）  **一、实验结果分析（包括数据处理、实验现象分析、影响因素讨论、综合分析和结论等）**  由图片不难看出，测试结果用例全部正常通过，通过率为100%。  **二、实验小结及体会**  这次实验是对上次的结对编程的项目做单元测试，在上学期的软件工程导论课程也学习了单元测试相关的理论知识，但是并没有进行编程学习，理解地还不够透彻，借助本次的实验，自己使用了Junit test进行了单元测试相关的编程以及相应的实际操作，既拓宽了知识面也增强了理解，最最重要的是通过单元测试的方式大致了解了一个企业中软件开发的基本流程之中最容易被人忽视的一个环节便是测试的环节，虽然说单元测试这项实验安排在自己的程序已经可以正常进行的基础上显得略有多余，但是我还是了解到了单元测试本身的强大与方便之处，也对本来一直以为的不太重要的测试方面有了显著的改观，更是自己对于一个实际上的软件编程有了许多的了解，为此我感到十分激动而欣慰。    **成绩评定表：**   |  |  |  |  | | --- | --- | --- | --- | | **序号** | **评分项目** | **满分** | **实得分** | | **1** | **实验报告格式规范** | **2** |  | | **2** | **实验报告过程清晰，内容详实** | **4** |  | | **3** | **实验报告结果正确性** | **2** |  | | **4** | **实验分析与总结详尽** | **2** |  | |  | **总得分** | **10** |  | | | | | | |

实验课程名称：\_\_软件工程基础实验\_\_\_\_

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验项目名称** | | **代码评审与程序性能优化** | | | **实验成绩** |  |
| **实 验 者** | | **张淳慷** | **专业班级** | **软件zy1802** | **组 别** |  |
| **同 组 者** | | **刘怡博** | | | **实验日期** | **2020.5.28** |
| **第一部分：实验预习报告（包括实验目的、意义，实验基本原理与方法，主要仪器设备及耗材，实验方案与技术路线等）**   1. **实验目的** 2. 了解代码审查的含义； 3. 了解如何对程序进行性能优化； 4. 掌握配置工具的安装与使用； 5. **实验内容及要求** 6. 针对前面实验中所完成的代码，进行代码评审(走查)和性能分析，从时间性能角度对   代码进行优化；   1. 练习代码评审的两个方面：静态分析、动态分析(profiling)； 2. 使用以下三个工具完成实验：   – Checkstyle （下载：http://checkstyle.sourceforge.net/）  – FindBugs （下载：http://findbugs.sourceforge.net/downloads.html）  – PMD （下载：https://sourceforge.net/projects/pmd/)   1. 按“实验一”的分组方式，两人一组，随机分配另一组的代码作为本组评审和分析的对象，实验期间不能与原作者进行沟通。 2. 实验环境   PC机、eclipse | | | | | | |
| **第二部分：实验过程记录（可加页）（包括实验原始数据记录，实验现象记录，实验过程发现的问题等）**   1. Checkstyle 2. 在preferences中对checkstyle进行配置，这里使用的是eclipse自带的Google代码规范      1. 对项目使用activate Checkstyle唤醒Checkstyle，使用check code with checkstyle对代码进行审查，检查代码是否有报错处。报错示例如下      1. 查看报错处并修改，直至无错。      1. 使用clear checkstyle violations去除标记，使用deactivate Checkstyle关闭checkstyle。 2. FindBugs 3. 配置项目的findBugs设置。      1. 显示bug区域及错误     此处提醒说改错操作会直接退出程序，而这就是这个功能的本意，故可以不用加以修改，而且说明设置的威胁侦测等级过高，应加以修改。   1. 修改过后，优化显示的bug直到无法继续优化。 2. PMD 3. 配置PMD      1. 运行PMD，得出依照规则查找出的缺陷。     Eg：    比如说此处显示说过长的代码缺乏注释的跟进，因为该类错误过多而且实际上并不存在影响运行的地方，故可以暂时不用对其进行修改。   1. 依据缺陷对代码进行优化。 2. visualVM   （1）VM配置如下：  （2）性能测试结果：      通过显示不难发现作为一个贪吃蛇的小游戏，其内存以及Cpu的占比只占了很小的一个部分，性能测试方面所展示出的数据也符合一个小体量的游戏所应该展现出来的水平。   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | 第三部分 结果与讨论（可加页）  **一、实验结果分析（包括数据处理、实验现象分析、影响因素讨论、综合分析和结论等）**   1. Checkstyle     根据报错信息进行修改，修改后规范性大大提高，直至无错,无错后的代码显示如下：     1. FindBugs   优化所有bug，程序安全性得到提升。，修改Bug后的程序显示如下：     1. PMD   根据我自己的查找因为PMD本身的默认识别逻辑过于严格，主要显示错误均集中在于没有添加注释或者是没有进行java.doc等的辅助性问题上，程序本身并没有出现有大的编码上的重大问题，仅仅只出现了有一些声明的变量没有进行调用的小的缺陷，对这些缺陷进行相应的更改后，程序显示如下：    **二、实验小结及体会**  这次实验分别学习了CheckStyle，FindBugs，PMD的安装和使用，他们都是通过eclipse的Install New Software来安装使用，然后需要进行配置，最后才能使用这些工具进行代码分析。这些工具的使用简单有效，十分方便。其次学会了使用Java监视和管理控制台对代码性能进行分析，通过观察相关的统计图像，得出了哪些因素的占用内存量大，或者CPU随时间的变化情况等。与此同时也使得我自己对于代码评审和程序优化有了更好的认知，不再像从前做算法题目一样只希望得出正确的结果而不管代码的美观等，这在合作编程中是极度有害的，也多亏这次实验自己得意纠正了自己不良的编程习惯。并且更深刻地体会到了结对编程的方便性，一个其他人视角的引入对于帮助自己跳出自己思维的局限性是十分有帮助的，不得不说这次实验着实是受益良多。    **成绩评定表：**   |  |  |  |  | | --- | --- | --- | --- | | **序号** | **评分项目** | **满分** | **实得分** | | **1** | **实验报告格式规范** | **2** |  | | **2** | **实验报告过程清晰，内容详实** | **4** |  | | **3** | **实验报告结果正确性** | **2** |  | | **4** | **实验分析与总结详尽** | **2** |  | |  | **总得分** | **10** |  | | | | | | | | |

实验课程名称：\_\_软件工程基础实验\_\_\_\_

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实验项目名称** | **UML建模工具的安装与使用** | | | **实验成绩** |  |
| **实 验 者** | **张淳慷** | **专业班级** | **软件zy1802** | **组 别** |  |
| **同 组 者** |  | | | **实验日期** | **2020年6 月4日** |
| 第一部分：实验预习报告（包括实验目的、意义，实验基本原理与方法，主要仪器设备及耗材，实验方案与技术路线等）  **一、实验目的**  1)学习使用EA开发环境创建模型的一般方法，理解EA界面布局和元素操作的一般技巧。  2)熟悉UML中的各种图的建立和表示方法。  3)掌握如何通过EA工具完成相关模型的建立。  二、**实验过程及要求**  1)学会安装、启动和关闭EA开发环境。  2)使用目标项目（Project）的向导创建一个项目。  3)了解EA界面的六大组成部分（菜单Menu、浏览器Browser、文档 窗口Document Window、图窗口Diagram Window、日值窗口Log Window和 工具栏。  4)在浏览器中实现如下操作：增加模型元素（包括角色、用例、类、 组件、图等）；查看现有模型元素；查看现有模型元素之间的关系；移动 模型元素；更名模型元素；访问元素的详细规范。  5)分别在图中增加、删除图元素和在浏览器中增加删除图元素。  三、**实验环境**  Windows10  第二部分：实验过程记录（可加页）（包括实验原始数据记录，实验现象记录，实验过程发现的问题等）   1. 实验过程 2. 下载安装包      1. 安装成功      1. 使用目标向导创建项目      1. 选择类型      1. 图中添加元素      1. 图中删除元素      1. 中元素的属性查看与更改功能及名称      1. 浏览器中添加及查看更名元素      1. 综合使用EA完成图的建模 | | | | | |
| 第三部分 结果与讨论（可加页）  **一、实验结果分析（包括数据处理、实验现象分析、影响因素讨论、综合分析和结论等）**  实验过程顺利，结果无误，成功完成实验要求并且利用EA工具完成了基本用例图的建模  **二、实验小结及体会**  通过本次实验初步了解了UML建模工具EA并且在实际中加以运用，之前做大作业时使用的UML建模工具实际上太过于简单，十分轻型而且所含有的功能十分简陋，但是 EA 的功能十分强大，而且复杂，目前很明显的感受到自己只是掌握了其基本功能，希望自己可以深入学习使用 EA 工具去了解其更加复杂的功能，再之后的实验中更加熟练地使用 EA 软件    **成绩评定表：**   |  |  |  |  | | --- | --- | --- | --- | | **序号** | **评分项目** | **满分** | **实得分** | | **1** | **实验报告格式规范** | **2** |  | | **2** | **实验报告过程清晰，内容详实** | **4** |  | | **3** | **实验报告结果正确性** | **2** |  | | **4** | **实验分析与总结详尽** | **2** |  | |  | **总得分** | **10** |  | | | | | | |

**实验五 UML系统分析与设计**

**----UML建模案例**

|  |  |  |
| --- | --- | --- |
|  |  |  |
| 实验5-1 | 用例图设计 |  |
| 实验5-2 | 类和对象图设计 |  |
| 实验5-3 | 交互图设计 |  |
| 实验5-4 | 状态图设计 |  |
| 实验5-5 | 活动图设计 |  |
|  |  |  |
|  |  |  |
|  |  |  |

**成绩评定表：**

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **评分项目** | **满分** | **实得分** |
| **1** | **实验报告格式规范** | **2** |  |
| **2** | **实验报告过程清晰，内容详实** | **10** |  |
| **3** | **实验报告结果正确性** | **4** |  |
| **4** | **实验分析与总结详尽** | **4** |  |
|  | **总得分** | **20** |  |

### 实验5-1 用例图设计

**一、实验目的**

掌握在EA中用例图的基本用法和使用技巧。

**二、实验内容与要求**

本实验给予系统的用例图的设计和实现。

需求描述如下：

某学校的网上直播系统主要包括如下功能：老师通过系统管理界面进入本节课的课堂，将课程，会议号等信息保存在数据库中并可以对其进行改动和删除，与此同时还可以进行提问、发出课件、点名、发布作业、投票等操作，这些操作结果将会通过业务层存入数据库中学生通过客户机浏览器根据学号和密码进入会议界面，在这里学生可以进行浏览课件、举手、投票、回答问题、签到、写作业的操作。同样，通过业务层，这些操作结果存入数据库中。

分析：

本系统拟用三层模型实现：数据核心层、业务逻辑层、和接入层。其中，数据核心层包括对于数据库的操作；业务逻辑层作为中间层对用户输入进行逻辑处理，在映射到相应的数据层操作；而接入层包括用户界面，如登录界面，用户选择界面等。

本系统涉及的用户包括老师和学生，他们是用例图中的活动者，他们的主要特征相似，都具有姓名和学号等信息，所以可以抽象出一个基活动者people，学生和老师均从people中统一派生。而数据处理系统则是另外一个活动者。

以老师添加课件事件为例：

1. 老师进入管理界面，用例开始
2. 系统提示输入密码
3. 老师输入密码
4. 系统验证密码
5. A1密码错误
6. 进入管理界面，系统显示目前所有可以进行的操作
7. 老师添加课件
8. 系统提示上传对应课件
9. 老师上传课件
10. 用例结束

其他事件：

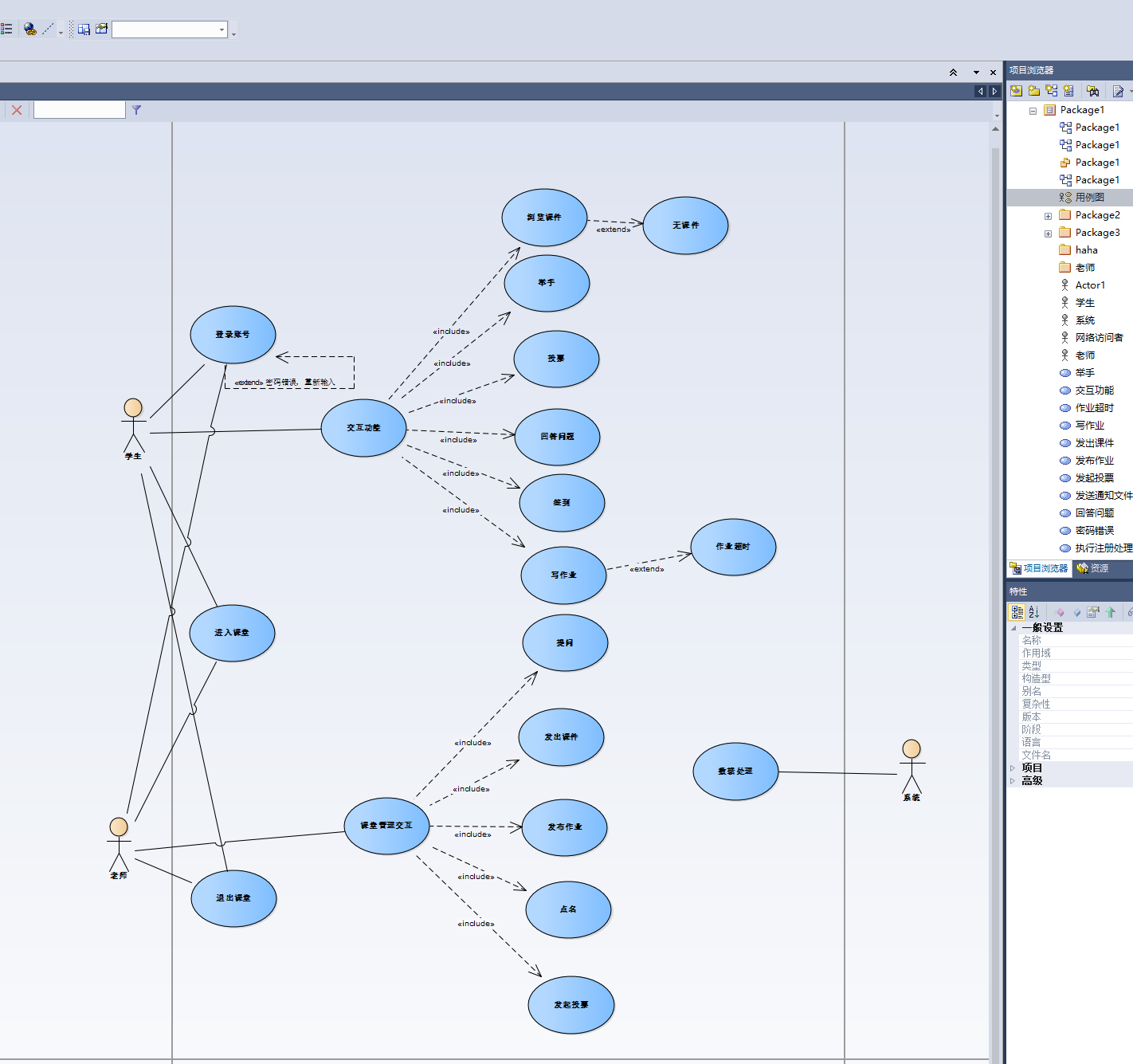
A1密码错误：

1. 系统提示再次输入
2. 用户输入
3. 三次输入错误系统自动关闭

其他事件答题情况相似，故不多做赘述

根据以上分析，绘制系统所有用例图。

**三、实验过程与结果**



**四、实验小结和体会**

老实讲当初做大作业的时候只负责去完成大作业的任务，并没有仔细地去思考用例图之中每个之间具体有什么关系关联，而通过这一次的实践，我明白了活动者不一定单指人，他可以是指系统以外的，用例图的边界是指系统内部的，同时还明白了用例的泛化，包含和扩展关系在实际应用层面之上的彼此的关系，尤其是深刻理解了泛化和包含关系之间的区别，也就是在用例的泛化关系中，所有的子用例都有相似的目的和结构，它们是整体上的相似。而用例的包含关系中，基础用例在目的上可以完全不同，但是它们都有一段相似的行为，它们的相似是部分的相似不是整体的相似。用例的泛化关系类似于面向对象中的继承，它把多个子用例中的共性抽象成一个父用例，子用例在继承父用例的基础上可以进行修改。但是子用例和子用例之间又是相互独立的，任何一个子用例的执行都不受其他子用例的影响。而用例的包含关系是把多个基础用例中的共性抽象为一个被包含用例，可以说被包含用例就是基础用例中的一部分，基础用例的执行必然引起被包含用例的执行。

### 实验5-2 类和对象图设计

**一、实验目的**

掌握EA中类图和对象图刻划系统的一般步骤和技巧，掌握类图和对象图中的各种基本概念的含义和表达方法。

**二、实验内容与要求**

根据实验5-1的问题描述，抽象出网上直播系统中的类，然后完成的类图。

1. 在系统中，涉及到的用户包括老师和学生，其主要特征相似，所以可以建立一个统一的基类people，而老师和学生则由people派生，创建他们三者之前的类图
2. 假设在系统中设计了以下类
3. 网页
4. 系统
5. 会议

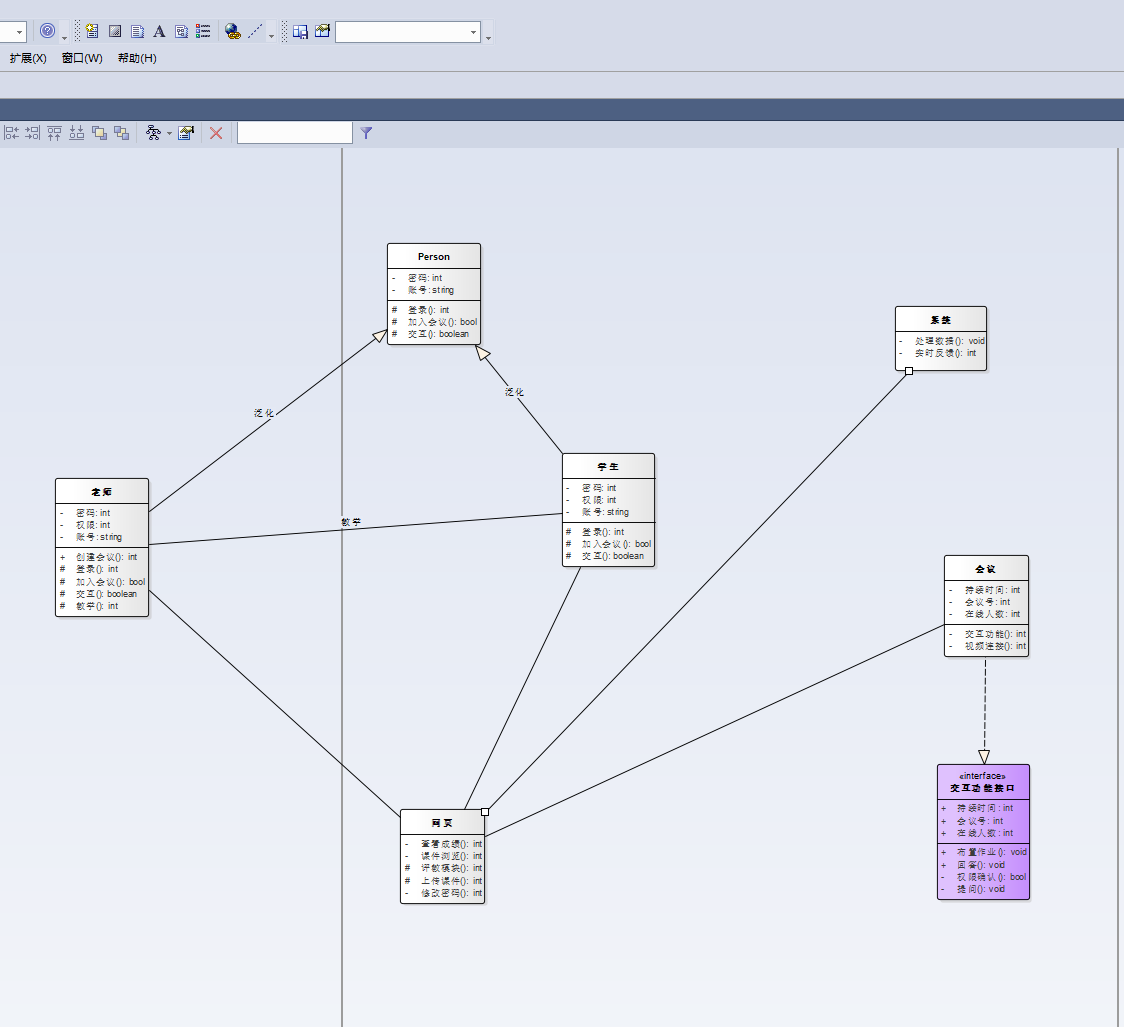
设计了以下角色：

1. 学生
2. 老师
3. 二者的父类people
4. 系统

他们的大致关系如下：

1. 角色学生和老师从people中派生
2. 学生和老师之间的交互通过网页决定
3. 会议表达出其会议号持续时间等，并且其交互功能还有相应的加护功能接口实现
4. 会议最终在网页上体现出来
5. 系统对网页上产生的各种数据进行处理不给出实时的反馈

**三、实验过程与结果**



**四、实验小结和体会**

进一步熟悉了对于类图的构建，也明白了画类图之前一定要提前分析类还有类与类之间的关系，确定他们的属性以及相应的操作，这样画起图来才能做到事半功倍，否则就需要不断的调整，十分的浪费时间。在理论上，深刻记住了类之间的关系，也即是依赖关系，泛化关系，关联关系以及实现关系使用时的注意之处，举例来说依赖关系表示的就是两个或多个模型元素之间语义上的连接关系。它只将模型元素本身连接起来而不需要用一组实例来表达它的意思。

这种定义利用通俗地理解成提供者的某些变化会要求或指示依赖关系中客户的变化，也就是说依赖关系将行为和实现与影响其他类的类联系起来。像这种通过实验加强自己的理论知识的方式自己着实是十分喜欢，正如那句名言所说的那样“实验是检验真理的唯一方法”。希望自己可以一直带着这份热情一直坚持下去。

### 实验5-3 交互图设计

**一、实验目的**

掌握交互图（序列图和协作图）绘制的基本方法和技巧，学会使用交互图描述用例实现的方法，掌握系统动态特性建模的技巧。

**二、实验内容与要求**

本实验以网上直播系统中的添加课件用例为例，来学习交互图的设计和实现。

为简化问题，避免不必要的麻烦，不考虑老师的登陆。假设老师已经成功登陆系统，选择的事件可以简化如下：

添加课件事件：

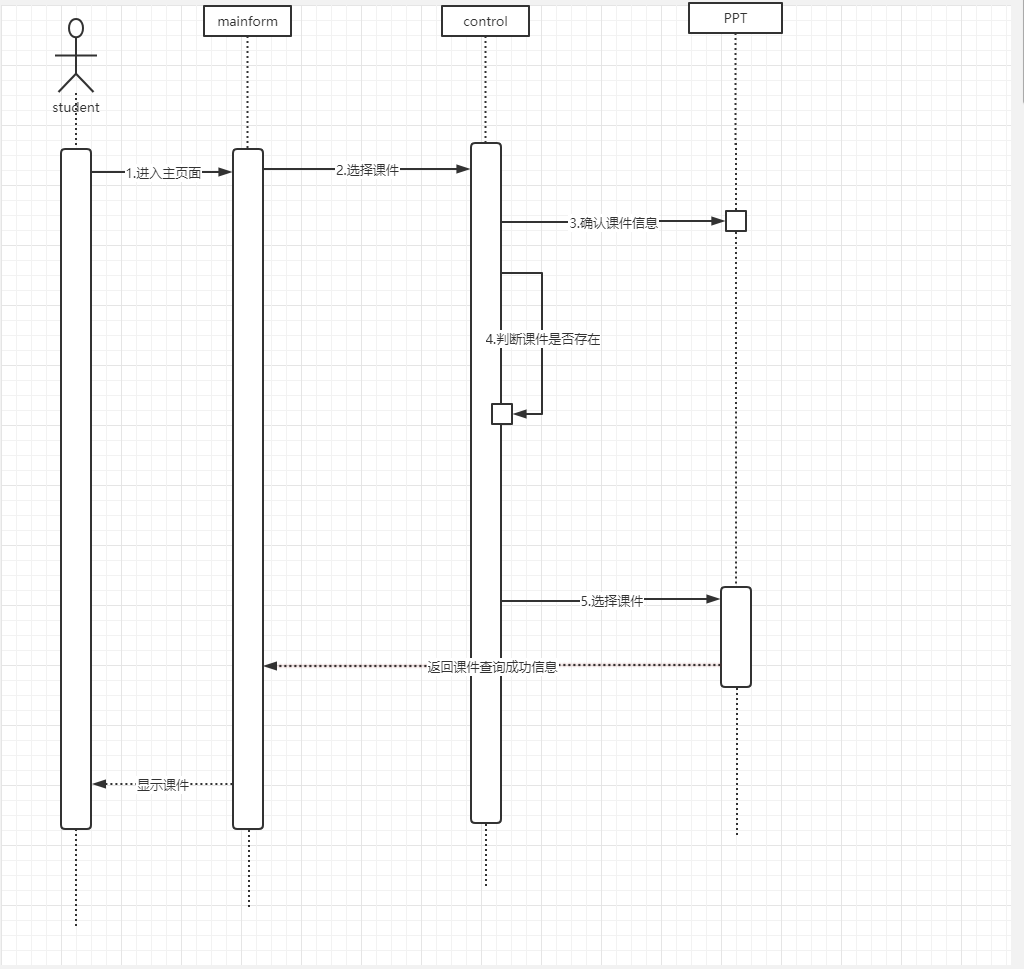
1. 进入系统主界面
2. 老师点击添加课件
3. 系统显示添加课件页面及其他要求
4. 老师添加课件
5. 系统验证课件是否可添加 A1课件不可添加
6. 系统提示课件添加成功
7. 用例结束

错误时间：

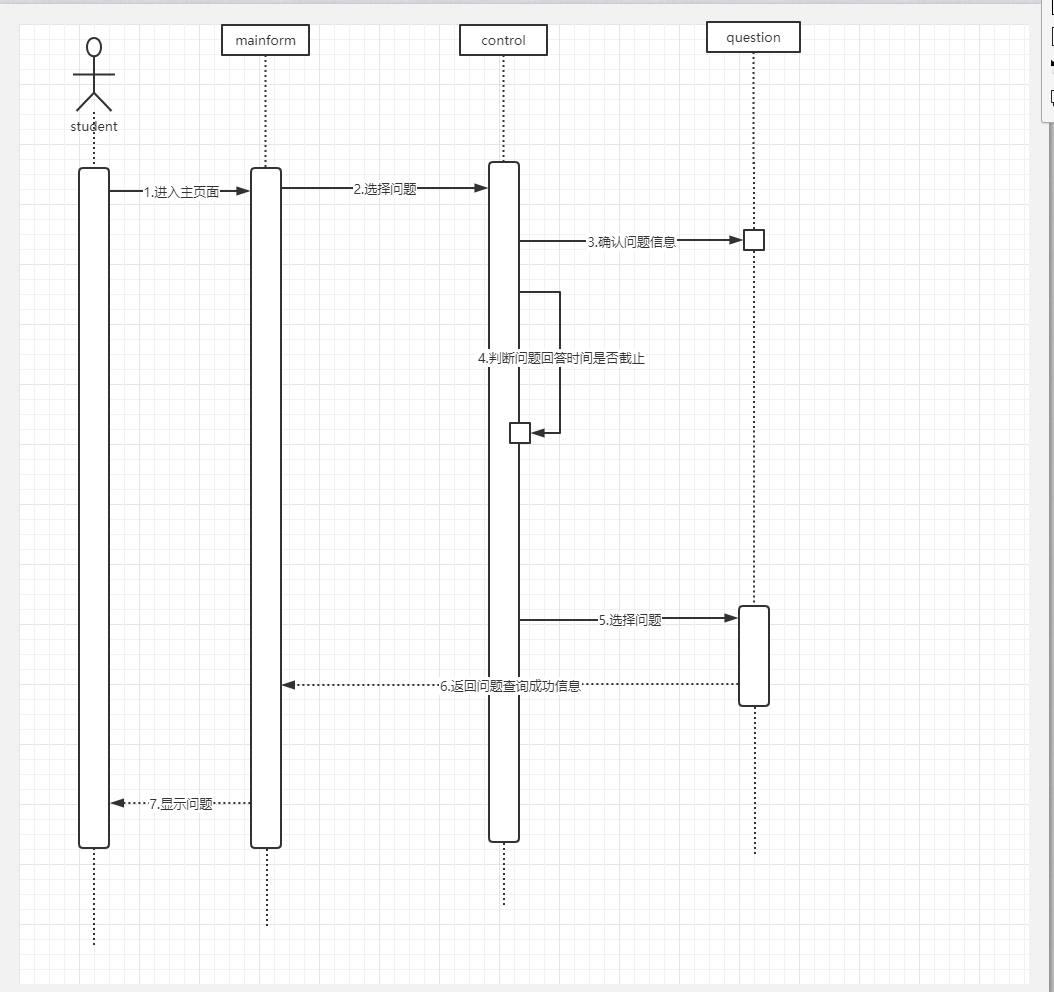
1. 系统显示课件添加失败原因
2. 老师重新添加课件
3. 重新验证直到成功
4. 转添加课件事件第6步

创建上述添加课件用例的交互图（序列图）。同时根据实验5-1的描述创建其他各用例的序列图。（因EA软件在作图工作中崩溃，故改用ProcessOn进行作图）

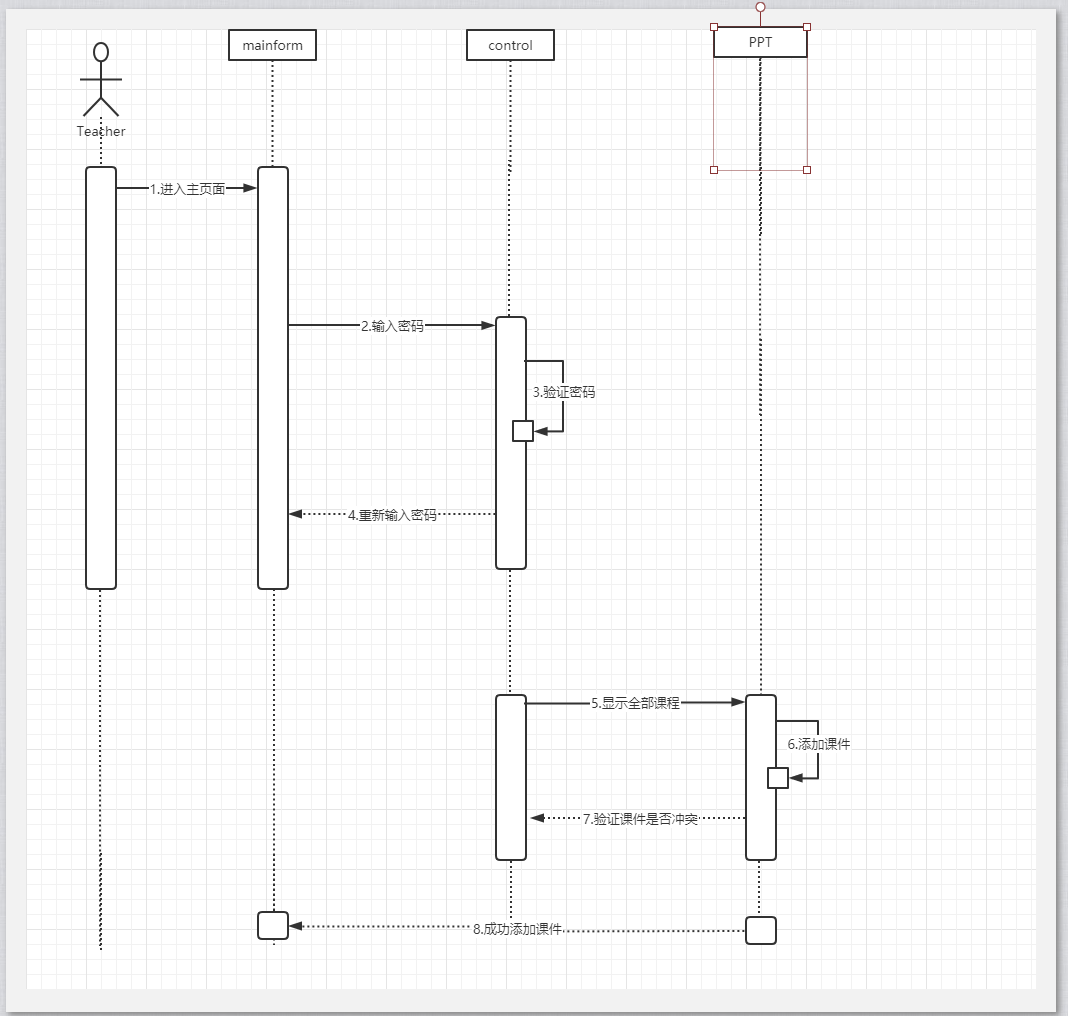
**三、实验过程与结果**



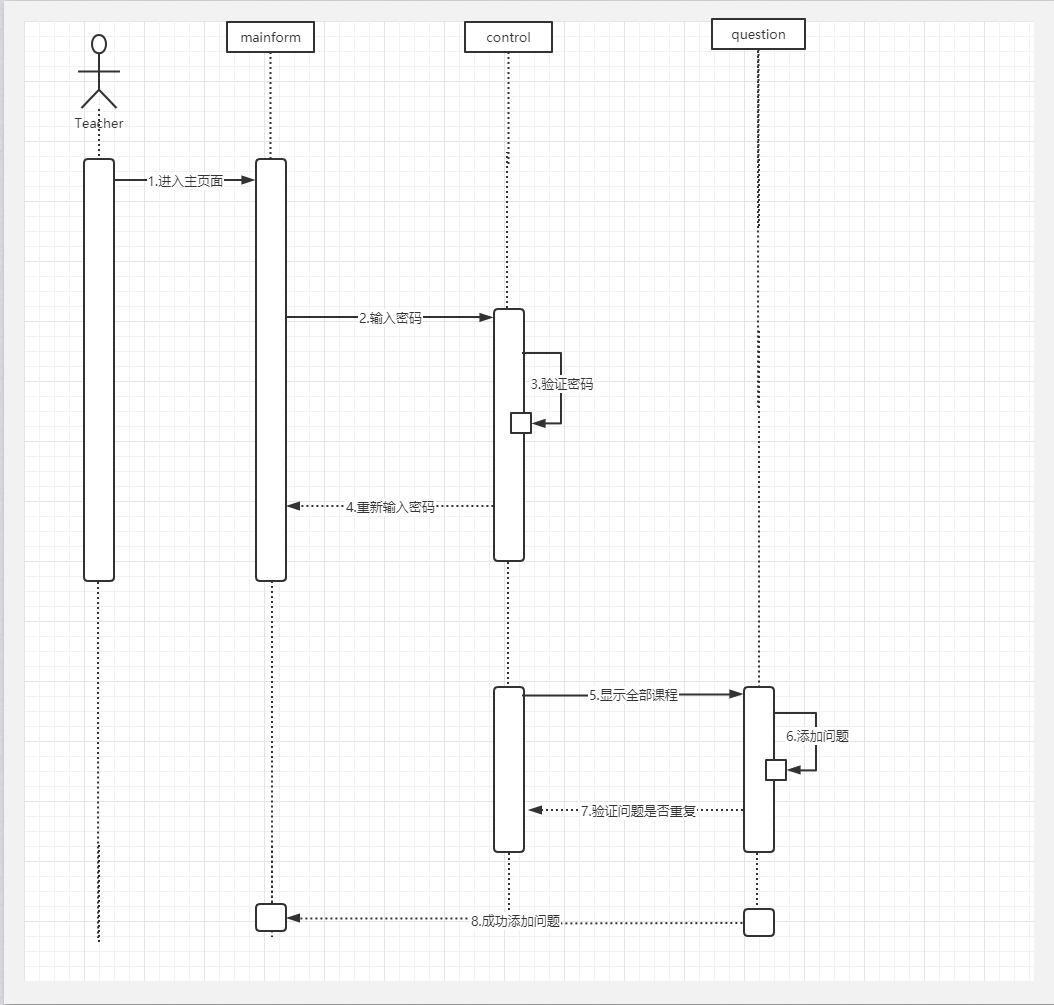
学生查找课件序列图



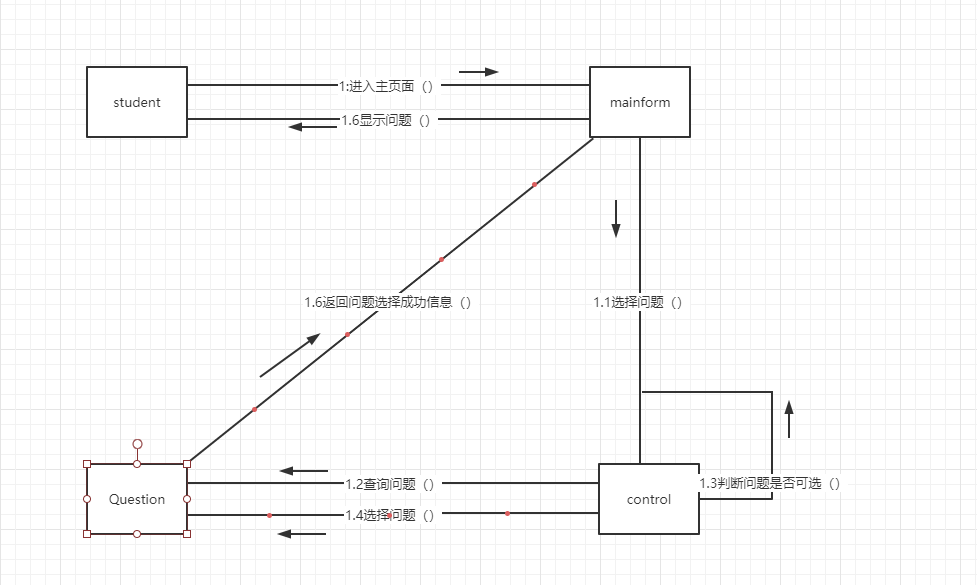
学生回答问题序列图



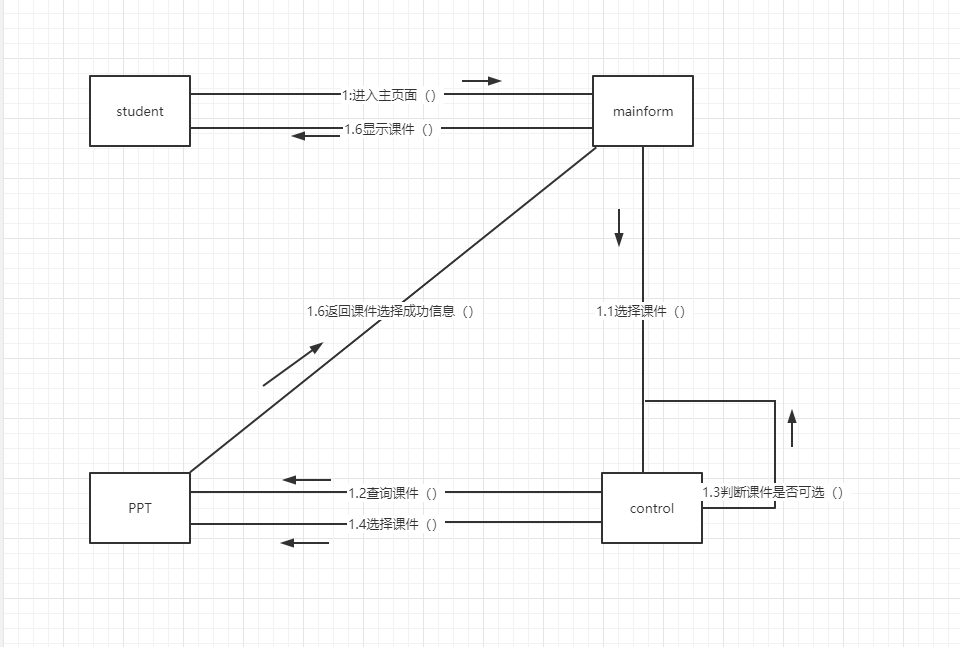
老师添加课件序列图



老师添加问题序列图



选择问题协作图



查询课件协作图

**四、实验小结和体会**

通过这次实验,我更清楚明白的感受到了协作图与序列图的强大之处，它使我的对于软件设计的思路更加的清晰明了，以序列图为例，他整个图就是详细表达出了一项功能实现的整个的具体的流程，从用户的登入使用直到最后信息的成功输出给用户看这整个本应内化在内部用户无法了解的过程通过这种图的形式简洁明了的展现在了面前，使得软件的设计工作可以更加顺利地进行下去，这样的知识实在是令我兴奋而激动，在理论知识上，通过画图我还了解了时序图与协作图之间的相同点和不同点，拿相同点来说，时序图和协作图都规定了消息发送对象 和消息接收对象的责任,每个接收对象都有消息对应的接口,这个接口方法由消息发送对象触发;时序图和协作图都支持消息;时序图协作图可以作为衡量系统耦合性的工具, 耦合性就是 系统中模型之间的依赖性,两个模型之间的依赖关系通过查看两个模型之间的消息数量和类型就可以看出,消息交互少的耦合性高。再拿不同点来说，**时序图**侧重描述对象的创建和撤销，新创建的对象放在对象生命线上对应的时间点,撤销的对象在结束的地方放一个 X 表示该对象不能再继续使用;**协作图**中创建撤销的概念不存在,对象时始终存在的,只能通过消息描述或约束 来说明对象的创建和撤销;**时序图**中的链没有表示出来,可以随意绘制消息,有些逻辑交互可能不会发生;**协作图**的消息是映射在链上的,消息和链平行放置,如果想要通过消息查看对象之间的关联使用协作图最好;时序图表现有激活和去激活，协作图上没有时间概念的描述,无法清晰地表示对象的激活和去激活。学习了这些知识的我感到激动不已，也希望自己可以一直带着这份热情继续下去。

### 实验5-4 状态图设计

**一、实验目的**

掌握状态图绘制的基本方法，理解状态图中各个要素的含义及表达。

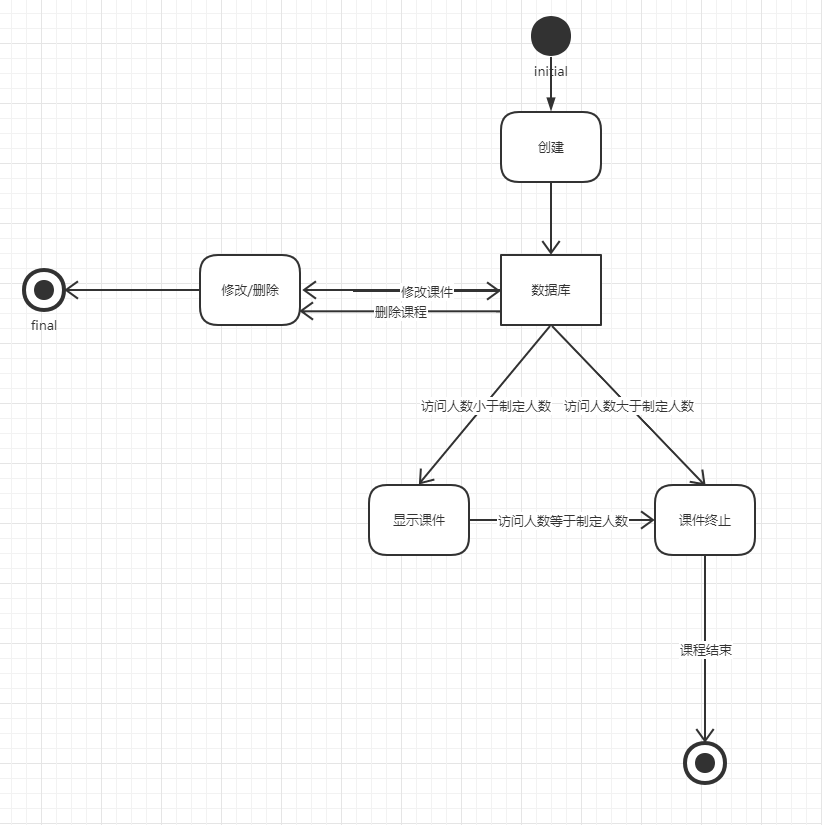
**二、实验内容与要求**

1. 考察课件类（对象）的状态变化过程。给出对象状态描述：

课件对象被创建，添加到数据库中。管理员（老师）可以删除修改其信息，在上课期间，传入该课件，如果访问人数大于制定的人数，则不允许访问。课程结束，课件的状态终止，无法再被查看。

1. 根据以上描述，绘制课程对象的状态图。

**三、实验过程与结果**



**四、实验小结和体会**

通过这次实验，我详细了解了状态图整体的创建流程以及他们所表达出的意义，对于之后的学习有很大的帮助

### 实验5-5 活动图设计

**一、实验目的**

掌握使用活动图描述各种复杂事件的逻辑。

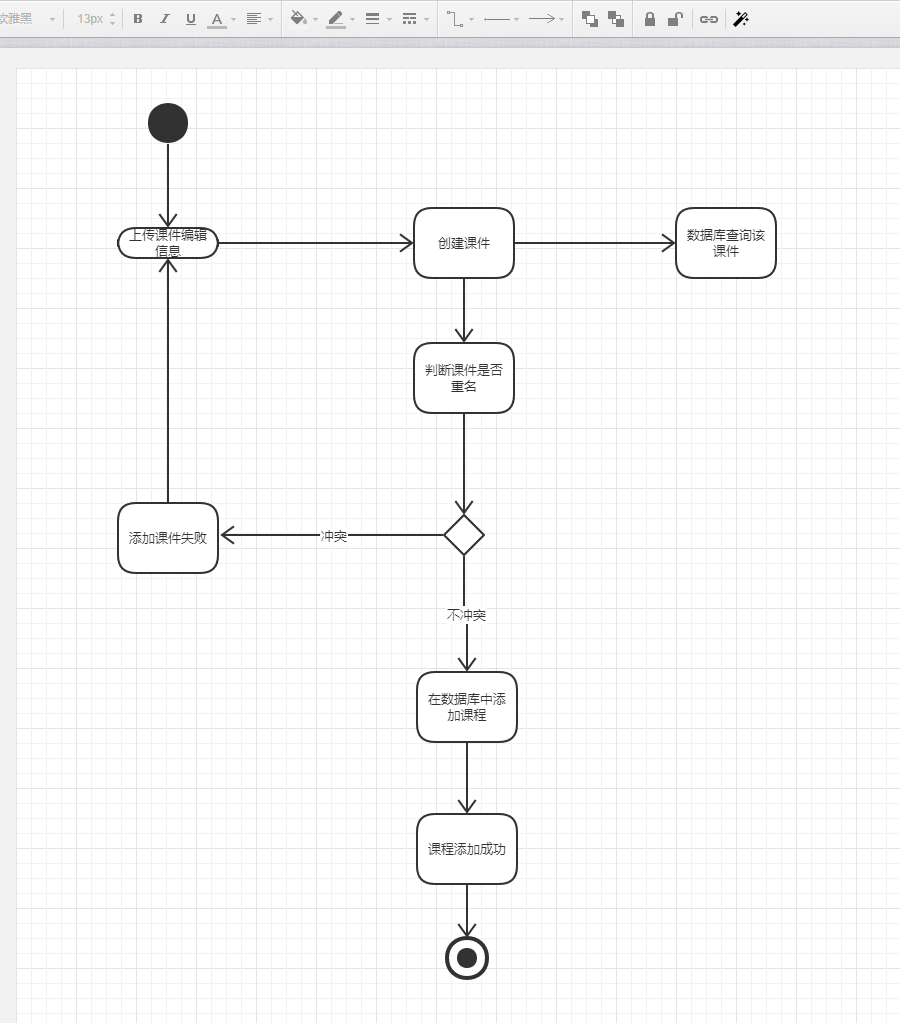
**二、实验内容与要求**

在用例图中，我们对添加课件用例事件做了详细分析（见实验5-2）。由于老师密码登录过程可以抽取出来，作为通用的流程，所以这里对添加课件事件稍作修改，将老师输入课件信息作为起始的活动，内容如下：

1. 老师上传课件
2. 系统验证是否与已有的课件冲突。A2有冲突
3. 系统添加课件，提示课件添加成功
4. 系统重新进入主界面，显示所有的课件
5. 用例结束

根据以上分析，创建添加课件完整的活动图。绘制活动图时注意各元素的表示方式。

**三、实验过程与结果**



**四、实验小结和体会**

活动图是状态图的另一种表现形式，活动图相比于状态图更接近于代码所想表达出来的意图，相对来说更像是一个流程图，通过查阅资料了解到，活动图着重于表现一个活动到另一个活动的控制流，是内部处理驱动的流程。一般一个活动结束后，自动就会转到下一个活动，而状态图则更强调一个状态到另一个状态的流程主要有外部事件的参与。通过了解发现了自己只是水平的进一步上升，我感到十分开心，也更有信心去学号这一门课。

实验课程名称：\_\_软件工程基础实验\_\_\_\_

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验项目名称** | | **系统测试** | | | **实验成绩** |  |
| **实 验 者** | | **张淳慷** | **专业班级** | **软件zy1802** | **组 别** |  |
| **同 组 者** | |  | | | **实验日期** | **2020/6/30** |
| 第一部分：实验预习报告（包括实验目的、意义，实验基本原理与方法，主要仪器设备及耗材，实验方案与技术路线等）  **一、实验目的**  （1）了解负载测试、压力测试等性能测试的概念。  （2）能使用常用工具 JMeter 进行性能测试并对根据测试结果进行性能分析。  （3）进一步掌握软件压力测试的常用方法。  **二、实验要求**  （1）根据实验指导书给出的关于JMeter的使用介绍，了解 JMeter测试软件的功能结构。  （2）并选择对一个已有的网站或自己事先设计好的动态或静态 网站（页面），进行测试和分析 | | | | | | |
| 第二部分：实验过程记录（可加页）（包括实验原始数据记录，实验现象记录，实验过程发现的问题等）   1. 安装Jmeter     （2）使用Jmeter进行不登录的正常测试  1.设置测试网站（百度）    2.进行基本的不登录测试    返回值为200，正常运行成功  （3）采用badboy进行压力测试（采用qq邮箱）  1．badboy脚本录制：    2．导出为.jmx，并打开jmeter，设置线程数等    3．添加监听器    4．开始压力测试  汇总报告：    聚合报告： | | | | | | |
| 第三部分 结果与讨论（可加页）  **一、实验结果分析（包括数据处理、实验现象分析、影响因素讨论、综合分析和结论等）**  通过聚合报告不难看出，总的来说qq邮箱网站从总共的6387个样本中不难看出其异常率略高，但是其吞吐率属于一个网站中所有的水平中相对较好的一档，所以总的来说qq邮箱的服务器还是十分先进的，撑住了这一次的压力测试。其综合表现十分优秀，属于正经的腾讯所有的网站系统。  **二、实验小结及体会**  老实讲在这么多次软件基础实验的过程中，经历了画图、结对编程等相对来说比较枯燥的内容之后，这次跟压力测试有关的内容着实是十分引起了我的兴趣，用线程去模拟用户去测试一个网站本身的水平，真的是一件十分有意思的事情，尤其是我选用的网站是qq邮箱的网站，现在一会想过来可能还有点后怕自己的线程会不会设置得调高导致其封我的ip，不过这个实验过程来讲尽力这么一个全新的之前从来没有经历过的实验是十分激动人心的，我也在这之中学会了许多与系统测试有关的知识概念，跟坚定了未来要走计算机这条录得决心，希望自己可以带着这份决心和热情已知走下去。    **成绩评定表：**   |  |  |  |  | | --- | --- | --- | --- | | **序号** | **评分项目** | **满分** | **实得分** | | **1** | **实验报告格式规范** | **2** |  | | **2** | **实验报告过程清晰，内容详实** | **4** |  | | **3** | **实验报告结果正确性** | **2** |  | | **4** | **实验分析与总结详尽** | **2** |  | |  | **总得分** | **10** |  | | | | | | | |

实验课程名称：\_\_软件工程基础实验\_\_\_\_

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **实验项目名称** | | **Git实战** | | | **实验成绩** |  |
| **实 验 者** | | **张淳慷** | **专业班级** | **软件zy1802** | **组 别** |  |
| **同 组 者** | |  | | | **实验日期** | **2020/6/30** |
| 第一部分：实验预习报告（包括实验目的、意义，实验基本原理与方法，主要仪器设备及耗材，实验方案与技术路线等）  **一、实验目的**  1）熟练掌握git的基本指令和分支管理指令；  2）掌握git支持软件配置管理的核心机理；  3）在实践项目中使用git/github管理自己的项目源代码； | | | | | | |
| 第二部分：实验过程记录（可加页）（包括实验原始数据记录，实验现象记录，实验过程发现的问题等）  **二、实验流程**  **I．git的操作**  **i.仓库创建与提交**  1.安装git（由于之前已经安装过故直接给出桌面上所显示出的快捷方式以及主界面）      说明（该版本号为2.27.0）  2.申请github账号（由于之前已经拥有故直接给出登录界面）  账号信息：    3.配置git    4．仓库的创建    5.将文件提交到仓库    6.生成ssh码    7.添加ssh密钥    8.同步远程仓库    9.推送成功    **ii.分支管理**  1.创建查看当前分支    2.对当前分支作出修改并提交，而后查看主分支    3.分支合并    4.删除dev分支    **iii.远程分支管理**  1.原仓库代码如下    2.推送分支      3.抓取分支    4.分支提交    5.分支提交结果    **II．Eclipse中git plugin的安装、配置及使用**   1. git plugin的配置     输入github上注册的对应的邮箱以及用户名     1. git plugin的使用   （1）将项目变为git项目（右击项目选择team中的 share projct，后选定文件夹，成功创建git项目）    （2）将项目提交到Github库（右击项目选择team中的commit，全选项目之后加上注释点击commit）    （3）使用push    这之中要先在github上创建一个新的库并将对应的URL复制到eclipse上的对应位置，后选择master分支点击add spec完成push操作  （4）提交结果（网址为<https://github.com/Za-ck/greedy-snake/tree/master/snake>） | | | | | | |
| 第三部分 结果与讨论（可加页）  一、实验结果分析（包括数据处理、实验现象分析、影响因素讨论、综合分析和结论等）  不难看出所有的实验均圆满完成，其中特别选择最基本的git的工作模式进行详细的分析，通过实验可以得出git实际上的工作流程就是如图所示    根据图即可知道其就是通过将工作区也就是本地的文件提交到暂存区中，再由暂存区整体提交给本地仓库后又仓库提交至远程仓库进行保存，而分支则可以理解为不同的开发流程所带来的结果，比如说要进行新功能开发却不确定其是否稳定则可利用分支的功能，带功能稳定后再同步至主分支上，总的来说，git确实是一个十分方便的工具  二、实验小结及体会  通过这次实验，我强烈地体会到了git的强大之处，其中最令人感到意外的就是其最为分布式版本控制系统的方便，因为其根本没有“中央服务器”，每个人的电脑上都是一个完整的版本库，这样，工作的时候，就不需要联网了，因为版本库就在自己的电脑上而且和集中式版本控制系统相比，分布式版本控制系统的安全性要高很多，因为每个人电脑里都有完整的版本库，某一个人的电脑坏掉了不要紧，随便从其他人那里复制一个就可以了。而集中式版本控制系统的中央服务器要是出了问题，所有人都没法干活了。同样的分布式管理系统带来的还有就是速度快以及灵活，还有便是分支冲突这个功能，它使得任意两个开发者之间可以很容易的解决冲突，并不会影响到大体地工程地进行，但是相应的git也有他的缺点，就是代码保密性差，一旦开发者把整个库克隆下来就可以完全公开所有代码和版本信息。所以说如果是本身开发者想要开源地去处理问题，集群众之智的话就可以采用git，但如果只是自己做一点小的应用开发或者是说进行很大型的需要保密的软件开发的话就不要使用git。  **成绩评定表：**   |  |  |  |  | | --- | --- | --- | --- | | **序号** | **评分项目** | **满分** | **实得分** | | **1** | **实验报告格式规范** | **2** |  | | **2** | **实验报告过程清晰，内容详实** | **4** |  | | **3** | **实验报告结果正确性** | **2** |  | | **4** | **实验分析与总结详尽** | **2** |  | |  | **总得分** | **10** |  | | | | | | | |

**《软件工程基础实验》成绩评定表**

|  |  |  |  |
| --- | --- | --- | --- |
| **姓 名** | **张淳慷** | **学 号** | **0121801100913** |
| **专业、班级** | **软件zy1802** | | |
| **成绩评定依据：**  结合平时考查，以实验实际操作的优劣、提交的文档和程序的规范性等作为考核依据。实验成绩按五级制评分，以学生实际实验工作能力的强弱作为评定成绩的主要依据。  最终成绩100% = 平时成绩20% + 实验成绩80% | | | |
| **成绩评定表：**   |  |  |  |  | | --- | --- | --- | --- | | **序号** | **评分项目** | **满分** | **实得分** | | **1** | **学习态度认真、遵守纪律** | **20** |  | | **2** | **实验一：结对编程** | **10** |  | | **3** | **实验二：单元测试** | **10** |  | | **4** | **实验三：代码评审与程序性能优化** | **10** |  | | **5** | **实验四：UML建模工具的安装与使用** | **10** |  | | **6** | **实验五：UML系统分析与设计** | **20** |  | | **7** | **实验六：系统测试** | **10** |  | | **8** | **实验七：Git实战** | **10** |  | |  | **总得分** | **100** |  | | | | |
| **最终评定成绩（以优、良、中、及格、不及格评定）** | | | |

**注：最终成绩以五级分制记。优（90-100分）、良（80-89分）、中（70-79分）、及格（60-69分）、60分以下为不及格**

**指导教师签字：**

**年**   **月 日**