**贵阳职业技术学院**

2016~ 2017 学年度第一学期考试（A）卷

出卷部门《信息科学系》 考试科目：《软件工程》

1.考试时间：100分钟。

2.请考生首先按要求在试卷装订线内填写您的姓名、考号、校区和所在教学单位的名称。

3.请仔细阅读各种题目的回答要求，在规定的位置填写您的答案。

4.不要在试卷上乱写乱画，不要在标封区填写无关的内容。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 总分 |  | 题号 | 一 | 二 | 三 | 四 | 五 |
| 核分人 |  | 题分 | 10 | 20 | 20 | 20 | 30 |
| 复查人 |  | 得分 |  |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| 得分 | 评卷人 | 复查人 |
|  |  |  |

|  |  |  |
| --- | --- | --- |
| 得分 | 评卷人 | 复查人 |
|  |  |  |

1. **单项选择题**（每题只有一个正确答案，共10分,每小题2分）

**错答漏答均不得分**

1. 关于软件项目的生命周期，下列哪项**不属于**开发阶段？（D）
   1. 需求分析
   2. 测试
   3. 概要设计
   4. 运行与维护
2. 关于软件开发模型，下列说法中**不正确**的是 （B）
   1. 常见的软件开发模型有瀑布模型、原型模型和螺旋模型等
   2. 面向对象模型并不是一种软件开发模型
   3. 现代软件开发模型强调尽早测试及快速迭代
   4. 不采用任何软件开发模型的软件开发项目往往随着项目复杂性的增大而很难成功
3. 关于软件系统的设计方法，下列说法中**不正确**的是（D ）
   1. 软件设计的过程总体可分为概要设计和详细设计
   2. 软件设计的大致经历了结构化的软件设计和面向对象的软件设计两个大的发展阶段
   3. 面向对象的设计模型也是由需求分析阶段建立的分析模型导出的
   4. 面向对象设计中模块设计的原则是高耦合、低内聚
4. 描述类之间的关系，最恰当的UML视图是（D ）
   1. 用例图
   2. 顺序图
   3. 状态图
   4. 类图
5. 关于软件工程标准，下列说法**正确**的是（D）
   1. 任何时候软件开发都只需要遵循一个软件工程标准就可以了
   2. 国际标准在软件工程的标准层次中处于最底层
   3. GB代表的是Great Britain，即英国国家标准
      1. ISO代表的是国际标准化组织，ISO/TC97专门负责与计算机有关的标准化工作
6. **不定项选择题**（每题**至少**有一个正确答案，共20分,每小题5分）

**漏答一个扣1分，错答一个扣2分，扣完为止**

* 1. 需求分析的任务包括下面哪几项？（AB）
     1. 建立分析模型
     2. 编写软件需求规格说明书
     3. 制定测试计划
     4. 设计系统原型
  2. 关于需求分析评审，下列说法**不正确**的有（ B）
     1. 需求分析评审应由用户和系统分析人员（开发人员）共同进行
     2. 需求分析评审不能包含在软件开发合同中
     3. 需求分析评审的一个功能是验证需求的一致性
     4. 需求分析评审需要经过规划、准备、召开审查大会、修改缺陷与重审等阶段
  3. 关于测试用例，下列说法**正确**的有（ AD）
     1. 测试用例={测试数据+期望结果}
     2. 测试用例={测试数据+期望结果+实际结果}
     3. 测试结果={测试数据+实际结果}
     4. 测试结果={测试用例+实际结果}
  4. 关于程序的编码风格，下列说法**正确**的有（B）
  5. 编码风格就是指使用结构化的程序语言编码
  6. 避免使用goto语句这种跳转语句是一种良好的编码风格
  7. 编码风格是通过程序设计语言的语法得到保证的，使用特定语言，编码风格就是统一的
  8. 程序的注释是不会被执行的，因而是无关紧要的，写不写都可以

|  |  |  |
| --- | --- | --- |
| 得分 | 评卷人 | 复查人 |
|  |  |  |

1. **填空题**（共20分,第一题8分，第二题12分）
2. 请根据下列标准的UML模型元素的图形，将其所对应的元素名称填充完整,并解释表示元素之间连接关系的符号含义：

类

➂属性

➃方法

➀状态

**依赖：**⑤一个元素以某种方式依赖于另一个元素

**关联：**⑥连接模型元素及实例

**泛化：**⑦表示一般与特殊的关系。一般元素是特殊元素的泛化

**聚集：**⑧表示部分与整体的关系，即部分元素是整体元素的一部分

1. 任何一个项目都是由若干任务组成，任务之间存在一定的相关性（或依赖性），请根据描述信息将下表中任务的相关性描述补充完整，或是解释某个相关性的含义：

|  |  |  |
| --- | --- | --- |
| 任务相关性 | 范例 | 描述 |
| 完成—开始（FS） | B  A | ➀\_只有任务A完成后任务B才能够开始\_ |
| ➁\_完成-完成\_ | A  B | 只有在任务A完成后任务B才能够完成 |
| 开始—完成(SF) | B  A | ➂\_只有任务A开始后任务B才能够完成\_ |
| ➃\_开始-开始\_ | A  B | 只有在任务A开始后任务B才能够开始 |

|  |  |  |
| --- | --- | --- |
| 得分 | 评卷人 | 复查人 |
|  |  |  |

1. **简答题**（共20分,每小题5分）

答中划线要点即可得分

1. 请简要说明一份好的测试缺陷报告应该包含哪些内容？

好的缺陷报告应该包括三大内容：1.预期结果与实际结果2.复现条件（软件的预置条件、复现步骤、软件版本、出现概率等）3.影响（软件版本、严重程度、解释说明等），以及测试者、测试日期等信息

1. 请简要说明软件文档在软件开发中的作用。

软件文档在软件开发中起着重要的桥梁作用：它是项目管理的依据、质量的保证、作为培训的参考、对软件维护提供支持、作为历史资料档案

（答中任意三点即可得分）

1. 程序的注释是程序的重要组成部分，请问良好的注释应该满足哪些要求？

应该在模块开头给出程序的整体说明，例如程序的标题、修改说明、主要算法及接口、数据说明、程序的位置、开发历史等。此外还应包含功能性注释，对程序中的难以理解的某段代码或算法进行解释

1. 请简要说明为什么需要制定软件工程标准，亦即软件工程标准化所带来的好处是什么？

软件标准的制定可以克服软件开发中程序设计个体化所导致的“软件危机”，规范软件开发过程，提升软件产品质量和开发效率，降低软件开发风险

|  |  |  |
| --- | --- | --- |
| 得分 | 评卷人 | 复查人 |
|  |  |  |

1. **应用题**（共30分）

现有一Web浏览器的开发项目。通过需求分析得到的主要功能为：

* + - 1. 浏览网页，保证所查看Web网页能够正确渲染；
      2. 缓存，能够将所浏览网站的原貌保存下来；
      3. 离线浏览，能够定义下载层数、下载文件类型以及是否跨网站下载；
      4. 网页内容分析，能够根据用户的兴趣提取网页中的部分内容

根据以上功能需求，得出该浏览器的具体行为模型包括但不限于：

1. 在URL栏中输入地址并按Enter后 ，在显示框中显示网页，同时在Cache文件夹中按网站的组织结构保存网页；
2. 用户单击超链接时，显示所链接页面，同时将该页面按网站结构缓存到Cache文件夹，同时在URL显示该页面URL地址；
3. 单击“Cache”按钮，则在显示框右侧打开显示Cache文件夹结构的树形目录，单击其中的网页文件名则在显示框中显示该网页；
4. 单击“Offline”按钮，则在浏览时从Cache中获取网页；
5. 要求尽可能使用本地Cache的网页以提高浏览速度。

请根据以上需求回答下列系统设计问题：

1. （5分）前端软件设计多采用MVC架构，即Model-View-Controller架构，请按照MVC架构对以下子模块功能进行划分，并归入相应的MVC模块中（第一个模块已填入）：

➀系统控制；➁网页获取；➂缓存管理；➃网页显示；⑤输入控制

Model

\_➀,\_➁➂\_\_

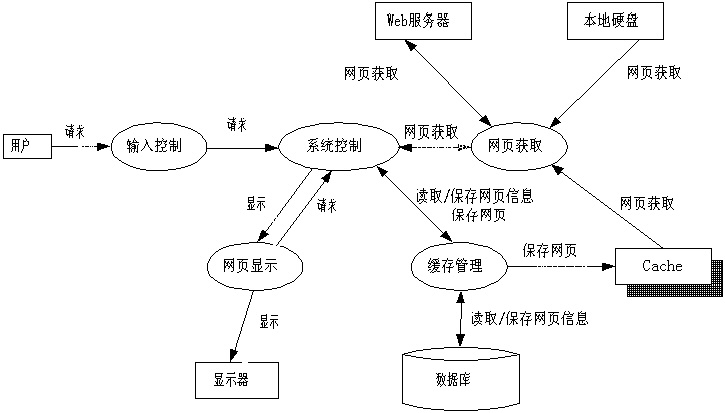
View

\_\_➃\_

Controller

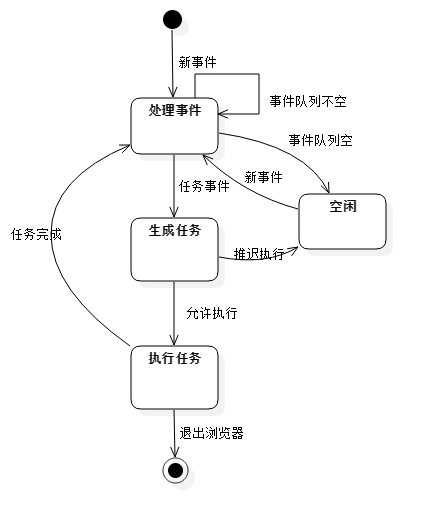
\_\_⑤\_

1. （15分）系统的对象关系图如下图所示：



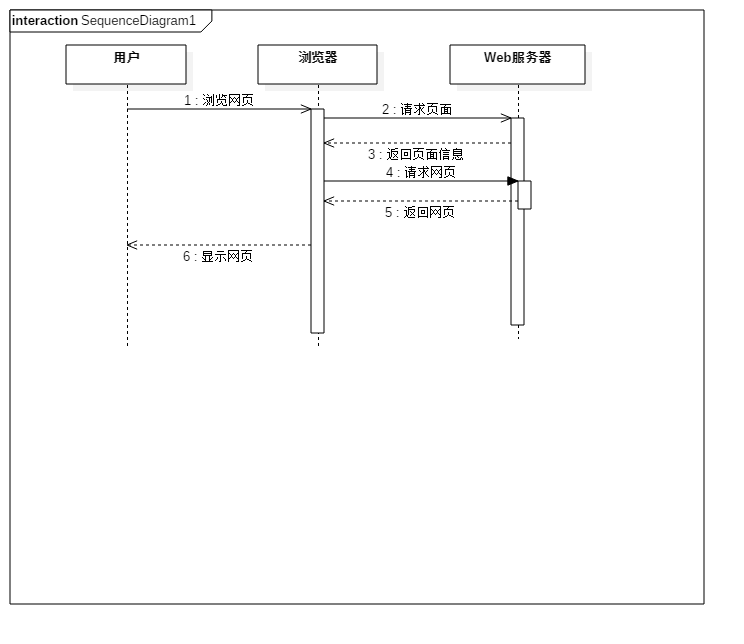
请根据前述的系统行为模型和对象关系图

* 1. 使用标准的UML模型绘制**系统控制类的状态转换图**。提示：任意时刻，系统只能处于某一种特定状态下，例如等待用户输入事件、处理事件、生成任务、执行任务，等等。



要点：状态用圆角矩形，开始和结束符号不能用错，表示状态迁移的线有箭头，有文字说明状态迁移条件

* 1. 使用标准的UML模型绘制**网页获取用况的顺序图**。提示：根据网页获取目标的不同(Web服务器、本地硬盘或是Cache中)，可能需要分成几个时序图来绘制。



本地硬盘及Cache的网页获取与Web网页获取类似，实体类对象不同而已。

1. （10分）根据以上分析，你认为应该如何设计系统中的各个实体类，需要遵循一些什么样的原则？请画出两个以上你设计的实体类的类图，说明他们之间的关系（聚合、关联、泛化还是依赖？）并解释你这样设计的原因（即你所遵循的原则）。

遵循的原则：

1. 单一原则，即一个类只提供一个功能
2. 高内聚、低耦合

实体类包括：输入控制类、网页获取类、网页显示类、缓存管理类等

缓存管理

保存网页

网页显示

显示网页

网页获取（接口）

获取网页

保存网页

显示网页

Web网页获取

获取Web网页

保存网页

显示网页

Cache网页获取

获取Cache网页

保存网页

显示网页

硬盘网页获取

获取硬盘网页

保存网页

显示网页

如图所示，网页获取类由三个子类继承，分别实现从本地硬盘、Cache(内存/缓存文件)及远端Web网站获取网页，这样实现可以使得这三个子类都符合单一原则，同时对系统内的其他模块（如网页显示类和缓存管理类）具有统一的接口（保存网页、显示网页），从而达到高内聚、低耦合的效果

（能够正确使用表示类关系的符号，如继承使用实线箭头，关联使用无箭头实线。能够正确识别接口方法）