第 1 章　上升到面向对象

选择题

1.A

2.A

3.B

简答题

1.　与传统结构化方法相比，面向对象技术的优势主要体现在哪些方面?

主要包括以下几个方面的与优势：

(1)沟通：在计算机中模拟现实世界的事和物；

(2)稳定：较小的需求变化不会导致系统结构大的改变；

(3)复用：提高质量，减少成本；

(4)改善软件结构，提高软件灵活性；增加可扩展性；支持增量式开发，支持大型软件开发等。

2.　什么是对象，什么是类，说明它们之间的区别和联系?

(1)对象是一个实体，这个实体具有明确定义的边界和标识，并且封装了状态和行为；

(2)类就是对象的抽象描述，这些对象共享相同的属性、操作、关系和语义。

(3)类是对象的抽象，而对象是类的实例，是具体的；通过类可以构造具体的对象。

3.　什么是抽象，如何进行抽象?

(1)抽象是揭示事物区别于其他事物的本质特征的过程；

(2)需要根据使用者的目的来进行抽象，强调使用者感兴趣的特征，而忽略那些不相关的特征。

4.　什么是封装，通过封装如何实现信息隐藏和数据抽象?

(1)封装是指对象对其客户隐藏具体的实现；

(2)通过封装，对象的私有数据不能被外界存取，实现信息隐藏，从而保证外界以合法的手段访问；

(3)通过封装，将数据访问过程抽象为对操作的调用，从而将数据抽象为行为。

5.　什么是分解，结构化分解和面向对象分解有何不同?

(1)分解是指将单个大规模复杂系统划分为多个不同的小构件；分解后的构件通过抽象和封装等技术形成相对独立的单元，这些单元可以独立地设计和开发，从而实现化繁为简、分而治之，以应对系统的复杂性，减少软件开发成本。

(2)结构化分解中，通过函数、模块等进行功能分解，实现模块化设计。通过耦合和内聚来判断分解的合理性，将系统分解为多个高内聚、低耦合的模块。而面向对象的分解则是在类和对象分解的基础上，进一步考虑类之间依赖程度、复用问题和稳定性等问题，进行合理的打包和分层，从而形成更加复杂的分解结构。

6. 什么是泛化，什么是多态，它们之间有什么关系?

(1)泛化是类与类之间一种关系，通过这种关系一个类可以共享另外一个或多个类的

结构和行为。

(2)多态在同一外表(接口)下表现出多种行为的能力；

(3)在对象技术中，　一般通过泛化关系建立类之间的抽象层次结构，再通过上层抽象多态调用底层实现。

7.　什么是分层，分层和分解有何不同?

(1)分层是指面向不同的目标建立不同的抽象级别层次，从而在不同的抽象层次对系统进行分解，进一步简化对系统的理解；

(2)分解一般是在系统的同一个抽象层对大的结构进行划分，而分层则是在不同的抽象层次上进行；大规模系统开发时，　一般首先通过分层技术建立不同的抽象层次，之后在各个层次上进行合理的分解。

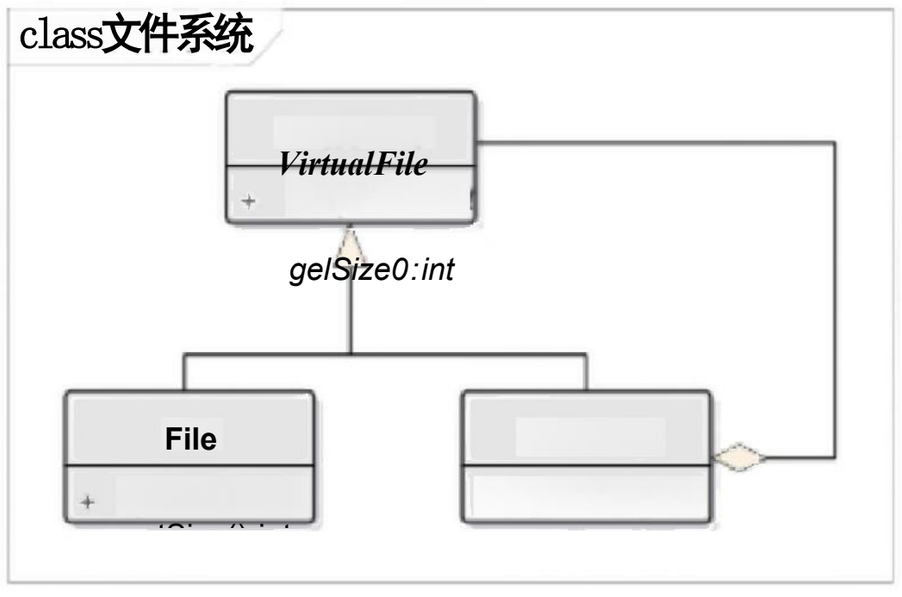
8.　什么是复用，在软件开发的哪些阶段可以进行复用?

(1)复用是借助于已有软件的各种有关知识建立新的软件的过程，以缩减软件开发和维护的成本；

(2)系统开发的各个阶段都可能涉及到复用，如代码复用、设计复用、架构复用、需求复用和领域复用。

应用题

1.　设计方案的核心是对文件 (File) 和目录 (Directory) 进行抽象，通过建立一个公共的抽象类(虚拟文件类 VirtualFile) 将两个概念联系起来，并通过聚合关系建立文件和目录之间的递归关系；而文件大小的计算，直接通过相应的类的操作来实现。设计类图如下图所示。



2.　本题为论述题，没有参考答案。读者可以选择任何一种熟悉的原则，结合某个具体的业务场景进行论述。比如系统中如何分层，如何复用第三方类库或以前的项目代码等等。

第 2 章　可视化建模技术

选择题

1. A

2. C

3. B

4.D

简答题

1. 通过建模技术，可以达到哪些目标?

(1)可视化：模型有助于按照所需的样式可视化系统；

(2)描述：模型能够描述系统的结构和行为；

(3)构造：模型提供构造系统的模板提高质量，减少成本；

(4)文档化：模型可以文档化设计决策。

2. 在系统建模过程中，需要遵循哪些基本原则?

(1)选择合适的模型；

(2)模型具有不同的精确程度；

(3)最好的模型是与现实相联系的；

(4)需要从多个视角创建不同的模型，单一的模型是不够的。

3. 哪些情况下，适合使用UML 进行系统建模?

(1)项目采用的 00 方法论；

(2)提高项目开发人员之间交流效率，准确抓住问题本质；

(3)系统的规模和设计都比较复杂，需要用图形抽象地表达复杂的概念，增强设计的灵活性、可读性和可理解性，以便暴露深层次的设计问题、降低开发风险。

(4)需要记录已成功项目、产品的公共设计方案，在开发新项目时可以参考、复用过去的设计，以节省投入，提高开发效率和整体成功率。

(5) 有必要采用一套通用的图形语言和符号体系描述组织的业务流程和软件需求，促进业务人员、软件开发人员之间一致、高效地交流。

4.UML 的语法结构使用什么方式来定义，如何定义?

(1)UML 语法结构采用UML 元模型来定义；

(2)主要是采用UML 类图描述各元素的抽象语法，采用约束机制和自然语言(文本)来描述模型语义。

5.UML 的语义结构主要包含什么内容?

UML的语义结构主要包括两类语义域

(1)结构语义定义了在建模域中关于个体的 UML 结构化模型元素的含义，这个含义可能只在某个特定的时间点是正确的，也称为静态语义。

(2)行为语义定义了在建模域中关于个体如何随着时间变化而做出不同行为的UML 行为模型元素，也称为动态语义。

6.UML 中的事物之间主要存在哪些基本关系?

UML 中的事物之间主要4类基本关系

(1)依赖是两个事物间的弱语义关系，表明两个事物之间存在着一种使用关系，其中一个事物(独立事物)发生变化会影响另一个事物(依赖事物)的语义。

(2)关联是一种强语义联系的结构关系，表明两个事物之间存在着明确的、稳定的语

义联系。

(3)泛化是一种特殊/一般关系，特殊元素(子元素)的对象可替代一般元素(父元素)的对象。

(4)实现是两个事物是之间的一种契约关系，其中的一个事物(箭头指向的事物)描述了另一个事物必须实现的契约。

7. 什么是构造型，UML 中如何利用构造型进行扩展?

构造型是 UML 的一种扩展机制，其作用是基于已有的建模元素扩展新的建模元素，可用于所有的 UML 模型元素。构造型的使用非常简单，只需要通过为已有元素设定一个构造型标记，以及相应的属性即可，也可以通过图标的方式区分不同的构造型。

8. 什么是外廓，如何利用外廓图扩展 UML 模型?

(1)外廓是基于UML 元素的子集为特定领域定义了UML 的一个特定版本，即定义了一组对 UML 已有模型的扩展和限定机制，以用于某个特定领域。这些扩展和限定机制包括：预定义的构造型、标记值、约束和基类等。

(2)外廓图是一种用于描述 UML 扩展机制的结构图，通过外廓图可以定义外廓包，以及特定的构造型、使用的元类、构造型和元类之间的扩展关系等内容，从而完成一系列的扩展。

9. 什么是 UML 架构中的视图，和 UML 图有什么区别和联系?

(1)视图可以理解为系统在某个视角的模型，每个视图面向不同的用户，提供不同的UML模型，以实现不同的建模目标。

(2)UML 图是特定的 UML 模型，视图由不同的 UML 图组成。根据视图所面向的用户和建模目标，选择不同的UML 图进行建模。

应用题

1. 不同的 UML 工具完成的图形可能会有细微的差别，本题答案主要检查核心建模元素的使用是否正确即可，不对细节要求过多。
2. 可以选择几种主流 UML 工具试用，分析其特点和主要使用场合。UML China 网站定期更新 UML 工具列表（http://www.umlchina.com/Tools/Newindex1.htm），感兴趣的可以根据该列表选择合适的工具。

第 3 章 业务建模

选择题

1. A
2. C
3. A
4. B
5. D
6. A
7. B

简答题

1. 本书讨论的 UML 分析设计过程主要包括哪几个阶段?

   主要包括6个阶段

(1)业务建模：采用软件建模方法分析和理解带开发的业务，描述业务流程；其目标是认识业务本质，该业务本质是后续用例建模的基础。

(2)用例建模：　采用 UML 用例建模技术描述软件需求，该需求模型将为后续用例分析提供输入。

(3)用例分析：　采用UML 用例分析技术分析软件需求，建立软件系统的分析模型。

(4)架构设计：　在系统的全局范围内，以分析模型为基础，设计系统的架构。

(5)构件设计：　根据架构设计的成果，将分析模型细化，设计系统构件的实现细节

(6)代码实现：　将系统构件映射到目标语言上。

2. 什么是业务建模，软件开发过程中为什么要进行业务建模?

(1)业务建模是一种建模方法的集合，目的是对现有业务进行分析和理解，从而建立相应的业务模型。

(2)业务建模有助于理解在业务领域中描述的事物是如何与软件领域中的事物相联系的，从而建立业务模型和系统模型之间的对应关系，以保证系统模型是能够满足业务需求的。

3. 什么是业务用例模型，业务用例模型主要包括哪些内容?

(1)业务用例模型是说明业务预期功能的模型，是业务建模阶段的核心模型，用于确定组织的各个角色和可交付工件。

(2)业务用例模型由业务用例和业务参与者构成，主要目的是说明客户和合作伙伴是如何开展业务的。

4. 什么是业务参与者，如何识别业务参与者?

(1)业务参与者代表了与业务有关的角色，此角色由业务环境中的某个人或物扮演。

(2)识别业务参与者的关键在于明确业务边界：业务参与者是在业务边界之外的、与业务进行交互的人或组织，它接受业务所提供的服务，并关注业务所产生的结果。

(3)在实际业务建模过程中，业务参与者可以是与业务进行交互的任何个人、组织、公司或计算机，可以从以下类别中查找参与者：客户、供应商、合作伙伴、潜在客户、政府、业务中为建模部分的人或组织。

5. 什么是业务用例，如何识别业务用例?

(1)业务用例展示了业务的外部视图，它确定了业务为了向业务参与者交付期望结果，需要执行什么流程；同时还确定了，在执行业务用例时，业务与业务参与者之间需要进行哪些交互。

(2)为了能够有效地识别主要的业务用例，可以从业务参与者的角度来考虑，即业务参与者通过业务用例从业务中获取价值。

(3)还可以从业务流程内部进行封装业务用例，研究业务内部的各类活动和流程，分析活动的目标，从而确定这些活动是为外部业务参与者提供怎样的服务，这些服务即可表示为业务用例。

(4)此外，还需要注意的是不要遗漏其他方面的业务用例，如一些支撑性业务流程(包括不直接使客户受益的活动)背后的业务用例。

6. 活动图中的动作节点什么条件下可以执行，有哪些种类的动作节点?

(1)当动作结点所有的对象流和控制流的前提条件都满足时，才创建动作的一次执行。

(2)根据动作执行所涉及的功能不同，可以划分为不同类别的动作，包括基本功能、行为调用、通信动作和对象处理等不同类型的动作节点

7. 什么是活动图中的控制节点，通过哪类控制节点可以进行并发行为建模?

(1)控制节点是一种特殊的活动节点，用于在动作节点或对象节点之间协调流程，表

示某一种控制动作。

(2)根据分叉和汇合这两个控制节点可以对并发执行和同步控制行为进行建模。

8. 活动图中的对象节点有哪两种表示方式，有何不同?

(1)有标准的对象节点和引脚两种表示方式

(2)标准的对象节点独立存在，而引脚则依附于动作节点，表示该动作的输入参数或输出值，是一种简化的表示方式，也可以转换为标准对象节点进行建模。

9. 活动图中的边可以设定哪些执行参数?

(1)可以为活动边设定执行条件、关联动作和权重等信息。

(2)执行条件为真时才能通过该活动边进入下一个动作关联的动作。

(3)关联动作表示在进入下一个动作节点之前需要提前执行的动作。

(4)权重规定了转移发生时输入对象的最小数目(常量或表达式),缺省为全部输入对象

10.什么是活动分区， 一般什么情况下对活动进行分区?

(1)活动分区用于识别具有相同特性的一组动作，这些动作被放入相同的区间。

(2)可以使用不同的分区规则进行分区，并没有严格的规范。

(3)在业务模型或需求中，往往按照组织机构的单位或系统角色进行分区， 一个单位或角色负责分区中所有节点的行为。而在设计模型中，可以按照不同的类(或构件)进行分区， 一个类(或构件)负责执行该分区中所有节点的行为。

11.什么是业务对象模型，业务对象模型主要包括哪些内容?

(1)业务对象模型从业务人员内部的观点定义了业务用例。该模型确定了业务人员以及他们处理和使用的对象(“业务类和对象”)之间应该具有的静态和动态关系

(2)业务对象模型主要包括：业务工人、业务实体和业务用例实现等内容。

12. 业务模型和系统模型之间有何区别和联系?

(1)业务模式建模当前业务现状，而系统模型则描述系统中业务实现模式。

(2)业务模型可以为系统模型中的用例视图和逻辑视图提供输入，还可以为系统架构提供一些重要的架构机制。

应用题

1. (1)申请人，(2)管理员，(3)公司前台

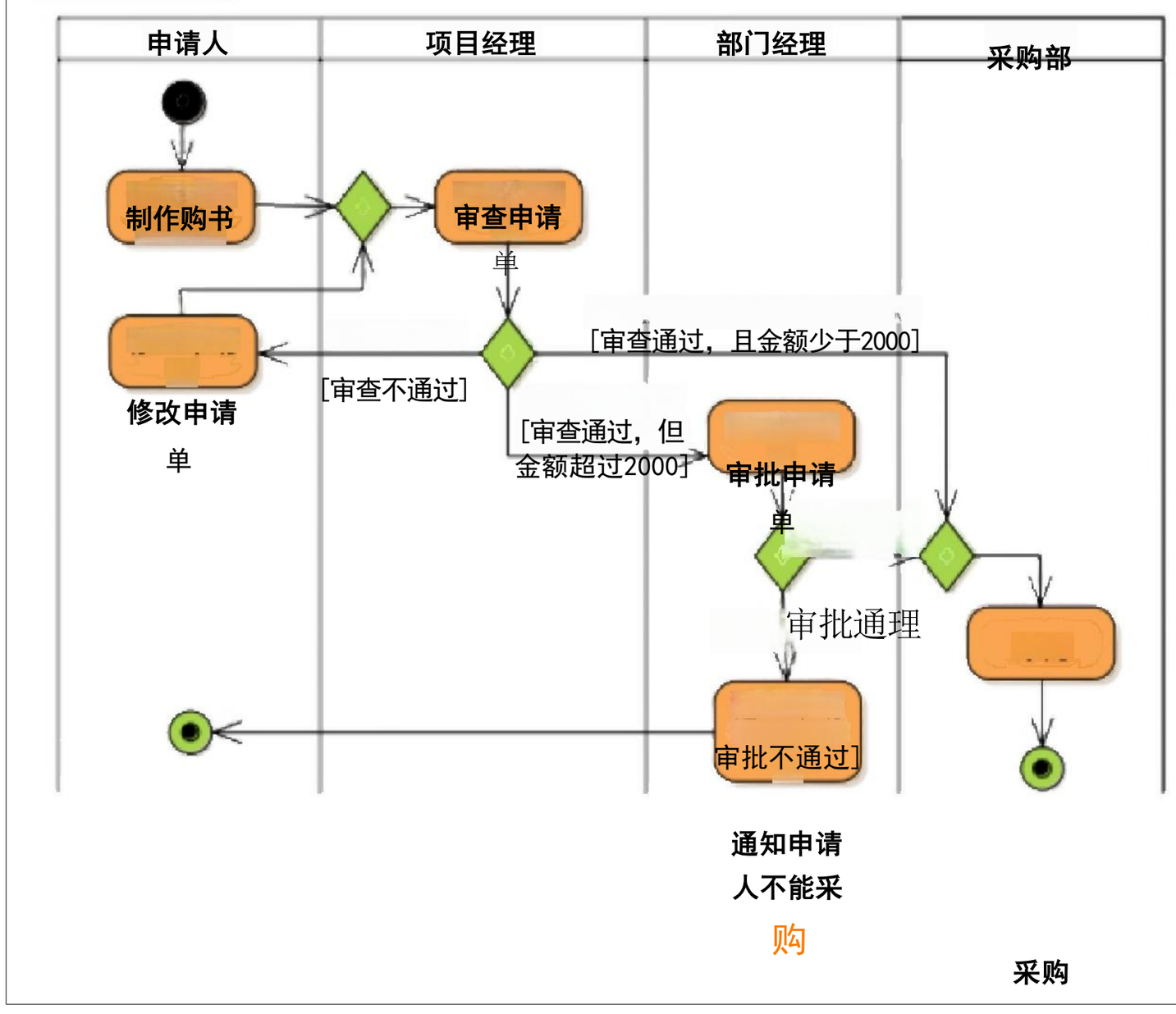
(4)需要预约，(5)需要取消预约，(6)取消接待室预约，(7)删除来访信息

(8)没有空余接待室，(9)有空余接待室，(10)预约不成功，(11)预约成功

(12)登记来访信息

2. 参考模型

act 图书采购申请



第 4 章　用例建模

选择题

1.C
2.C
3.C
4.D
5. A
6.A
7. D
8.A

简答题

1. 什么是需求，有哪些类型的需求?

需求是客户可接受的、系统必须满足的条件或具备的能力。RUP 中将需求分为 FURPS+多种类型，其中：

(1) 功能性 (Functionality): 详细描述了系统必须有能力执行的动作，通过详细说明所期望的输入和输出条件来描述系统行为；

(2) 可用性 (Usability): 人为因素(审美学、易学性、易用性)和用户界面、用户文档、培训资料的一致性；

(3) 可靠性 (Reliability): 主要包括故障的频率和严重性、输出结果的精确性、故障平

均时间 (Mean-Time-To-Failure,MTTF)、 故障恢复能力和程序的可预见性；

(4)性能 (Performance): 在功能性需求上施加的条件，如需求详述了交换率、速度、有效性、准确性、响应时间、恢复时间和内存使用，同时还加上了必须执行某个活动的条件；

(5)可支持性 (Supportability): 综合了可扩展性、适应性和耐用性等方面的能力，以及可测试性、兼容性、可配置性和其它在系统发布以后维持系统更新需要的质量；

(6)“+”号指还包含其它类型的需求，这些需求包括：设计约束、实施需求、接口需求和物理需求等。

2. 在业务建模之后，如何寻找业务改进点?哪些业务改进点可能会作为需求?

可以从以下四个方面来寻找业务的改进点

(1)流程控制：该业务涉及到复杂的控制流程，并在多个用户或部门之间流转；手工的信息流转方式难以满足业务需求，而且容易出错；

(2)复杂业务逻辑：某些活动涉及到一些复杂的业务逻辑，手工完成有很大的难度或工作量过大；

(3)使用业务对象：某些活动主要是对业务对象的操作，手工方式难以保证操作方式的合法性，并难以记录操作的历史等信息；

(4)自动化业务：某些业务要求定期、实时进行处理，其操作过程也不涉及复杂交互，手工方式难以保证按时完成。

对于每一个业务改进点，明确是否是为了达到远景目标的需要，如果是则作为软件需求而存在，并把相应的模型转化为系统模型；如果不是则不作为需求而存在，可能作为一项潜在的需求考虑，也可能直接抛弃。

3. 什么是系统参与者，识别参与者的主要要点包括哪些?

系统参与者代表了以某种方式与系统交互的人或事。更直观的说，参与者是指在系统之外，透过系统边界与系统进行有意义交互的任何事物。识别参与者的主要要点包括：

(1)系统外：参与者不是系统的组成部分，处于系统的外部；

(2)系统边界：参与者透过边界直接与系统交互，参与者的确定代表系统边界的确定；

(3)系统角色：参与者是一个参与系统交互的角色，与使用系统的人和职务没有关系；

(4)与系统交互：参与者与系统交互的过程是系统所需要处理的，即系统职责；

(5)任何事物：参与者通常是一个使用系统的人，但有时候也可以是一个外系统或外部因素、时间等外部事物。

4. 什么是系统用例，获取用例的主要要点包括哪些?

系统用例就是支持参与者与系统交互并达成参与者使用系统的目标，它由一组用例实例构成，而用例实例是系统执行的一系列动作，这些动作将生成特定参与者可观测的结果值。识别用例的主要要点包括：

(1)可观测：用例描述的是参与者与系统的交互，而不是系统内在的活动；因此用例的定义也应该只关注系统对外所体现的行为，或者说用例它止于系统边界。

(2)结果值：每个用例都会对外界参与者产生一个有价值的结果。

(3)系统执行：用例所产生的结果值是由目标系统所生成的。

(4)由参与者执行：用例的识别和定义都是从参与者的角度出发的，以参与者的视角获取和命名用例。

5. 什么是涉众，涉众和参与者有何区别和联系?

用例的涉众是指受用例所代表的业务影响的(或者说与当前用例有利益关系的)系统内外部人员或组织。由普通的人或部门来承担的参与者一般都是涉众。外部系统、时间等不是涉众，因为它们不是人或者组织，没有利益影响；不过当有外系统参与者时，那些外系统的用户往往会作为当前用例的涉众存在。

从涉众的角度来看，用例实际上是涉众之间所达成的契约，并以参与者为达成特定目标和系统交互的方式演绎。把用例比作一台戏，参与者和系统就是这台戏的演员，而涉众则是观众，戏的好坏由观众来评价。

6. 什么是用例的前置条件和后置条件，它们有什么作用，定义时需要注意什么?

前置条件是指用例在执行之前必须满足的条件，它约束用例开始执行前系统的状态。作为用例的入口限制，前置条件阻止参与者触发该用例直到满足所有条件。

后置条件是指用例执行完成之后系统的状态。当用例存在多个事件流时，可能会对应多个不同的后置条件。利用后置条件，有助于确保涉众理解执行用例后的结果。

在定义前置条件和后置条件时需要注意，只有在用例的使用者将这些条件视为附加价值的时候才使用，而且它们均要求是系统可以感知的(或者说检测到的);此外，前置条件还要求是在用例执行前就可以感知的。

7. 什么是用例的事件流，描述事件流是需要注意什么?有哪几种事件流，它们之间有何区别和联系?

用例的事件流是指参与者和系统交互的过程。在事件流描述时并不需要将这个完整的交互过程都表示出来；只需要描述需求部分，即用户需要什么,系统给出什么样的结果。其次，事件流的描述要使用户和开发人员互相理解用例的功能，需要注意以下几个方面的问题：

- ◆ 使用业务语言：使用用户平时所使用的语言进行描述。
- ◆ 重点描述参与者与系统交互的信息。
- ◆ 不使用[例如]、[等]这样的不清晰的表达。
- ◆ 不要过多的考虑界面细节。
- ◆ 不要描述系统内部处理细节，要描述从系统外部所看到的活动。
- ◆ 要明确描述用例的开始和结束： 一般事件流的第一句话表明该用例在何时如何开始；最后一句话表明用例的结束，有时可以不用显示的说明用例结束。

一个用例可能会存在多个独立的事件流。其中一条最核心的事件流称为基本事件流，其它的事件流则为备选事件流。

基本事件流又称为用例的主路径，是指在最一般的情况下，那些用例发生的路径。它通常用来描述一个理想世界，也就是说没有任何的错误发生。

备选事件流代表该用例处理过程中的一些分支或异常情况，它一般从基本事件流的某个步骤中分离出来。

8. 用例的补充约束主要包括哪些内容?如何描述补充约束?

用例重点在于描述功能需求，但对于系统来说，还存在很多功能之外的东西，比如非功能需求等，还有其它的一些诸如数据项的定义、业务规则、设计约束等内容。这些内容统称为补充约束。

与特定用例相关的补充约束，作为该用例文档中一部分来描述；而全局性的补充约束，单独形成一份独立的补充约束文档。在具体描述补充约束时， 一般采用自然语言来描述，但针对不同类型的补充约束，可以有不同的描述方法。如针对数据需求，可以采用数据字典的方式来描述；针对业务规则，可以使用决策表、OCL 等特定的工具或语言来描述。

9. 用例模型中，可以定义哪几种用例关系，它们有何不同?

用例模型中，用例的关系主要包括包含关系、扩展关系和泛化关系。

包含关系表示某个用例(基用例、主用例)中包含了其它用例(被包含用例、子用例)的行为。它提供了从两个或多个用例行为中提取公共部分的能力，把这些公共部分放到某个单独的用例中，通过包含关系来引用这些公共行为。

扩展关系是指某个用例(基用例、主用例)在特定情况下无法进行处理，而把这些行为委托给其它用例(扩展用例、子用例),表示该行为被扩展了。它的提出是为了将基用例的

一些特殊情况分离出来，在保持基用例本身相对完整的情况下(即一般情况都能处理)来处理这些特殊行为。

用例之间的泛化表明了一种继承层次，通过这种继承层次，特化的用例继承泛化用例的全部属性和行为，并参与泛化用例的各种关系。通过用例之间的泛化关系可以达到更高层次的需求复用，在泛化用例中描述通用行为，而特化用例继承这些通用行为，并在适当的地方进行特化，以处理具体的业务。

10.什么是扩展点，扩展点有什么作用?

扩展点是指在基用例中定义的特定条件，每个扩展用例都至少与一个扩展点相关联。当基用例满足了这些特定条件后，就会触发相应的扩展用例来为基用例提供附加行为。

11.有哪些用例的分包策略，　一般如何进行用例分包?

有几种用例分包的策略：

(1)基于业务主题的分包，按照用例所处理的业务领域不同，将面向不同业务主题的用例放在不同的包中。

(2)按照参与者分包，即相同参与者参与的用例放在同一个包里面，而不相关的参与者的用例放在不同的包。

(3)基于开发团队的分包，即结合开发团队的特点，将由同一个开发团队完成的用例放在同一个包中。

(4)基于发布情况的分包，即将在不同发布周期中发布的用例放在不同的包中，而将需要同时发布的用例放在一个包中。

在选择分包策略时，　一般首先结合业务特点按照业务主题进行分包(即每个包代表一个主题),再综合考虑开发团队和发布情况

12.如何对用例进行分级，高优先级的用例有何特征?

对用例分级并没有统一的标准，需要结合项目自身的业务特点以及开发团队的技术特点来综合考虑。　一般来说，高级别的用例是那些对系统架构有重要影响的用例，这些用例体现了系统的核心价值，也将成为后续分析设计的重点。

一般根据用例的重要性、复杂程度、风险等各种因素进行分级。具体来说，存在以下特征的用例一般具有较高级别：

(1)对系统架构有重要影响的用例。

(2)体现系统核心业务流程的用例。

(3)存在开发风险的用例。

(4)涉及新技术或者需要创新的用例。

(5)能够尽快投入使用并带来直接经济效益的用例。

应用题

1. 建议结合具体的实践项目讨论具体的方法，　一般都会涉及到收集资料、访谈、开会等基本的方法。
2. 主要的错误包括：

(1)参与者“每月初”改为“时间”

(2)普通用户和注册用户：如果有泛化就应去掉注册用户与浏览文章关联，或去掉泛化关系

(3)<<extend>>　的箭头方向画反了

(4)前置条件应去掉“且有要评论的文章”,因为用例启动前无法检测

(5)基本事件流2应去掉，不是该用例路径(因为登录已经作为前置条件了)

(6)缺少用例“申请开通blog”

3.

(1)时间

(2)管理已录制的节目

(3)预约录制

(4)extend

(5)电子节目指南系统

4.

(1)注册

(2)查询商品

(3)订购

(4)库存管理系统

(5)货物配送系统

5.[综合案例：考勤系统]

参见“考勤系统参考答案和评分标准”

6.[综合案例：医院预约挂号系统]

参见“医院预约挂号系统参考答案和评分标准”

第 5 章　用例分析

选择题

1.B

2.C

3.A

4.C

5. B

6. (1)B (2)A (3)D (4)D

7. (1)D (2)A (3)D

8. (1)C (2)A

简答题