

第一章(面向对象设计)课后习题

单选题

1. 如果想对一个类的意义进行描述,那么应该采用(C).
 - A.标记值-----p36(通常用来扩展UML构造块的特性,比如版本信息控制)
 - B.规格描述---p34(模型元素实例需要附加相关的规格说明来添加模型元素的特性)
 - C.注释
 - D.构造型-----p36(用来扩展UML词汇)
2. (A)模型的缺点是缺乏灵活性,特别是无法解决软件需求不明确或者不准确的问题.
 - A.瀑布模型
 - B.增量模型
 - C.原型模型
 - C.螺旋模型

多选题

3. 建立对象的动态模型的步骤有(ABCD)
 - [A].准备脚本
 - [B].确定事件
 - [C].构建状态图
 - [D].准备事件跟踪图
4. 软件的开发模式有(ABCD)
 - [A].瀑布模型
 - [B].xp开发模型
 - [C].喷泉模型
 - [D].构件开发模型
5. 下列关于类和对象的关系说法正确的是(ABC)
 - [A].有些对象是不能被抽象成类的
 - [B].类给出属于该类的全部对象的抽象定义
 - [C].类是对象集合的再抽象
 - [D].类是用来在内存中开辟一个数据区,存储新对象的属性

填空题

6. (UML)是面向对象技术领域内占主导地位的标准建模语言,它统一了过去相互独立的数十种面向对象的建模语言共同存在的局面,形成了一个统一的,公共的,具有广泛适用性的建模语言.
7. 类的定义要包含(名称),(属性),(操作).
8. 面向对象程序的三大要素是继承,封装,多态.
9. 面向对象的系统分析要求确立三个模型是功能模型,动态模型,动态模型.

简答题

(1).试叙述类和对象的关系.

类是具有相同或相似结构,操作和约束规则的对象组成的集合,而对象是某一类的具体化实例,每一个类都是具有某些共同特征的对象抽象.类与对象的关系就如模具和铸件的关系,类的实例化结果就是对象,而对一类对象的抽象就是类.类描述了一组有相同特性和相同行为的对象.

(2).简要叙述面向对象的概念.

面向对象技术是一种以对象为基础,以事件或消息来驱动对象执行处理的程序设计技术.从程序设计方法上来讲,它是一种自下而上的程序设计方法,从部分到全局.另外,它以数据为中心,类作为表现数据的工具.整体而言具有一定的稳定性.

(3).面向对象设计的原则.

- A. 模块化
B. 抽象
C. 隐藏
D. 低耦合
E. 高内聚

(4).软件的开发模式有几种? 各种的优缺点.

模式名称	优点	缺点
瀑布模型	有利于人员组织及管理,有利于开发大型项目	不能在开发过程中及时得到反馈,有较大项目风险; 缺乏灵活性; 完全确定用户需求是不太可能的事.
xp开发模型	策略简单,开发周期短.全程迭代增量开发,软件质量有保证.适应用户变化需求.	对小组成员合作要求较高.对大型项目比较耗时
喷泉模型	容易实现活动的迭代以及无间隙性	各个阶段有重叠,不利于管理.对文档要求高,审核难度大
构件开发模型	复用性强,提高效率.允许同时开发,可维护性强	缺乏组装结构标准,风险大,需要开发人员的等级较高

第二章

单选题

1. UML中的事务包括结构事物,分组事物,激活事物和(D)
- A. 实体事物

◦ B. 边界事物

◦ C. 控制事物

◦ D. 动作事物
2. UML中的四种关系分别是依赖,泛化,关联和(C)
- A. 继承

◦ B. 合作

◦ C. 实现

◦ D. 抽象

3. 用例用来描述系统子啊事件做出响应时采取的行动.用例之间是具有相关性的.在一个“订单输入子系统”中, 创建新订单和更新订单都需要检查永不账号是否正确.name,用例创建新订单,更新订单与用例检查账号之间是(包含)关系.
- A.包含
 - B.扩展
 - C.分类
 - D.聚集
4. UML中静态视图的可视化表达式的图主要包括(D)
- A.状态图
 - B.用例图
 - C.对象图
 - D.类图
5. 下列关于状态图的说法中,正确的是(D)
- A.状态图是UML中对系统的静态方面进行建模的五图之一
 - B.状态图是活动图的一个特例,状态图中的多数状态是活状态
 - C.活动图 and 状态图是对一个对象的生命周期进行建模,描述对象随时间变化的行为
 - D.活动图强调对有几个对象参与的活动过程进行建模,而状态图更强调对单个反应型对象建模

填空题

1. UML主要包含四种关系,分别是依赖,泛化,关联实现
2. 从可视化的角度来对UML的概念和模型进行划分,可将UML的概念和模型分为视图,图,模型元素
3. 物理视图包含两种视图,分别是实现视图和部署视图
4. 常用的UML扩展机制分别是构造型,标记值和约束.
5. UML的通用机制分别是规格说明,修饰和通用划分.

简答题

1. 在UML中定义面向对象的事物有哪几种.

A. 行为事物
B. 分组事物
C. 结构事物
D. 注释事物

2. 请说出构件的种类.

A. 源代码构件
B. 二进制构件
C. 可执行构件

3. 请说出视图有哪些种类.

A. 静态视图
B. 用例视图
C. 交互视图
D. 实现视图
E. 状态机视图
F. 活动视图
G. 部署视图
H. 模型管理视图

4. 请说出视图和图的关系.

包含和被包含的关系.

5. 简述UML的通用机制

通用机制作用：在UML图中添加各种描述信息，完善UML的语义表达。
包括：规格说明，修饰和通用划分。

第六章习题

单选题

1. 在ATM自动取款机的工作模型中（用户通过输入正确的用户资料,从银行取钱的过程）,下列不是参与者的是(D)
 - A.用户
 - B.ATM机存款管理员
 - C.ATM机取款管理员
 - D.取款
2. (B)不是构成用例图的基本元素
 - A.参与者
 - B.泳道
 - C.系统边界
 - D.用例
3. 下列不是用例间主要关系的是(C)
 - A.扩展
 - B.包含
 - C.依赖
 - D.泛化
4. 对于一个电子商务而言,(C)不是合适的用例.
 - A.用户登录
 - B.预定商品
 - C.邮寄商品
 - D.结账
5. 下列对系统边界的描述不正确的是(D)
 - A.系统边界是系统与系统之间的界限
 - B.用例图中的系统边界是用来表示正在建模的系统边界
 - C.边界内表示系统的组成部分,外界外表示系统的外部
 - D.我们可以使用Rose绘制用例中的系统边界

填空题

1. 参与者（Actor）,用例（UseCase）以及它们之间的关系构成的用于描述系统功能的动态视图称为(用例图).

- 2. 用例图的组成要素是(用例),(参与者)和(关系).
- 3. 用例中的主要关系有(扩展),(包含)和(泛化).
- 4. (用例的粒度)指的是用例所包含的系统服务或功能单元的多少.
- 5. 用例图中以实线方框表示系统的范围和边界, 在系统边界内描述的是(组成部分), 在边界外描述的是(系统外部).

简答题

1. 试述识别用例的方法

- A. 分析参与者开始
- B. 根据参与者来确定系统的用例
- C. 了解参与者如何使用系统, 需要系统提供什么样的服务.

2. 用例之间的三中关系各用在啥场合

关系	场合
包含	多个用例用到同一段的行为,将这段行为抽象为一个用例,其他用例包含它
扩展	处理异常或者构建灵活的系统框架,有利于降低系统的复杂度,提高系统性能
泛化	当系统的一个或者多个用例在行为,结构和目的方面存在共性时,可使用泛化

3. 在设计系统时,绘制的用例图多一些好还是少一些好,为什么?

视系统复杂度决定。
对于相对较简单的系统,可以相对用的少一些用例图。
对于较复杂的系统,具体情况具体分析,在尽可能保证用例图易于理解前提下决定用力的大小和数目。

4. 为何在系统设计时要使用用例图及其对用户有何帮助?

用例图是从软件开发需求到分析的第一步。
显示了系统的用户和用户希望提供的功能,有利于用户和开发人员实现沟通。
借助用例图,减少了大量交流上的障碍,便于问题达成共识。

5. 用Rose创建用例图的步骤.

- A. 识别参与者
- B. 创建用例
- C. 创建用例之间的关系

第七章(类和对象图)

单选题

- 1. 类图应该画在EA的(A)图形中.
 - A.UML Structural
 - B.UML Behavioral
 - C.Use Case View
 - D.Logical View
- 2. 类通常可以分为实体类,(C)和边界类.
 - A.父类
 - B.子类

- C.控制类
- D.祖先类

3. 对象特性的要素是(D).

- A.状态
- B.行为
- C.标识
- D.属性

4. 下列关于接口的关系说法,不正确的是(D).

- A.接口是一种特殊的类
- B.所有接口都是有构造型《interface》的类
- C.一个类可以通过实现接口支持接口所指定的行为.
- D.在程序运行的时候,其他对象不仅需要依赖于此接口,还需要知道该类关于接口实现的其他信息

5. 下列关于类方法的声明,不正确的是(C).

- A.方法定义了类所许可的行动
- B.从一个类创建的所有对象可以使用同一组属性和方法
- C.每个方法应该有一个参数
- D.如果在同一个类中定义了类似的操作,则它们的行为也应该是类似的

填空题

1. 对象图中的(对象)是类的特定实例,(链)是类之间关系的实例,表示对象之间的特定关系.
2. 类之间的关系包括(依赖)关系,(泛化)关系,(关联)关系和(实现)关系.
3. 在UML的图形表示中,(类)的表示法是一个矩形,这个矩形有三个部分构成.
4. UML中类元的类型有(类),(接口),(数据类型)和(构件).
5. 类中方法的可见性包含三种,分别是(private),(public)和(protect).
6. 类的职责指的是对该类的所有对象所具备的那些相同的属性和操作共同组成的(功能)或服务的抽象

简答题

1. 类图的组成元素有哪些?

类的:

- A. 名称
- B. 属性
- C. 操作
- D. 职责
- E. 约束
- F. 注释

2. 对象图由哪些部分组成

由对象和链组成.

3. 简述使用类和对象图的原因.

在面向对象设计中,类和对象的图形表示法是关键建模技术之一.
有效的对业务领域和软件系统建立可视化的对象模型.

4. 简述类图和对象图的关系和异同

名称	组成	异同
类图	类名; 属性; 操作	名称栏只包含类名; 属性栏定义了所有属性的特征; 类中列出了操作类使用了关联连接. 类是一类对象的抽象, 不存在多重性.
对象图	名称和属性	名称栏包含“对象名: 类名”; 属性栏定义了属性的当前值. 不包含操作内容. 使用链进行连接, 链中包含名称和角色. 对象可以有多重性

第八章(系列图)

单选题

1. UML中有四种交互图,其中强调控制流时间顺序的是(A).
- A,序列图
- B,协作图
- C,定时图
- D,交互概述图
2. 在序列图中,返回消息的符号是(B).
- A,直线箭头
- B,虚线箭头
- C,直线
- D,虚线

多选题

1. 序列图的构成对象有(ABCD).
- [A].对象
- [B].生命线
- [C].激活
- [D].消息
2. 在序列图中,消息编号有(AD).
- [A].无层次编号
- [B].多层次编号
- [C].嵌套编号
- [D].顺序编号
3. 下列关于序列图的说法,正确的是(ABCD).
- [A].序列图是对对象之间传递消息的时间顺序的可视化表示.
- [B].序列图从一定程度上更加详细地描述了用力表达的需求,并将其转化为进一步的,更加正式层次的精细表达
- [C].序列图的目的在于描述系统中各个对象按照时间顺序的交互过程
- [D].在UML的表示中,序列图将交互关系表示为一个二维图.其中,横轴是时间轴,时间沿竖线向下延伸.纵向代表了在协作中各独立对象的角色.

填空题

1. 在UML的表示中,(系列图)图将交互关系表示 为一个二维图.其中横向是时间轴,时间沿竖线向下延伸.纵向代表了在协作中各独立对象的角色.
2. 消息的组成包括(发送者),(接收者)和活动.
3. (激活)是对象操作的执行,它表示一个对象直接或通过从属操作完成操作的过程.
4. (生命线)是一条垂直的虚线,用来表示序列图中的对象在一段时间内的存在.
5. 序列图中对象的表示形式使用包围名称的(矩形框)来标记,所显示的对象及其类 的名称带有(下划线),两者用冒号隔开.

简答题

6. 简述系列图的用途

确认和丰富一个使用语境的逻辑表达.
细化用例的表达
有效地描述如何分配各个类的职责以及各类具有相应职责的原因.

2. 请描述系列图创建的步骤.

- A. 确定系列对象
- B. 创建对象.
- C. 创建生命线.
- D. 创建消息
- E. 销毁对象.

3. 简述项目开发中使用系列图的原因及其作用.

对交流当前业务如何进行很有用.
一个业务级的系列图能被当成一个需求文件使用,为实现一个未来系统传递需求.
项目需求阶段,分析师能够通过一个正式层的表达,把用例带入下一个层次.
设计阶段,架构师和开发这能使用该图,挖掘出系统对象间的交互,这样充实整个系统设计.

4. 说明对象图中销毁对象的方法.

销毁对象表示生命线的结束.
用“x”来进行标识
选中 [x] 按钮,此时光标变成“+”符号
单击欲销毁对象生命线.

第九章(协作图)

单选题

1. 关于协作图的描述, 下列(B)不正确.
 - A.协作图作为一种交互图, 强调的是参加交互的对象的组织
 - B.协作图是顺序图的一种特例
 - C.协作图中有消息流的顺序号
 - D.在Rose工具中, 协作图可在顺序图的基础上按F5键自动生成
2. UML中, 对象行为是通过交互来实现的, 是对象间为完成某一目的而进行的一些列消息交换.消息序列可用两种类来表示, 分别是(D).
 - A.状态图和顺序图

- B,活动图和协作图
- C,状态图和活动图
- D,顺序图和协作图

3. UML交互图中, 强调对象之间关系和消息传递的是(D).

- A,顺序图
- B,交互图
- C,定时图
- D,协作图

多选题

1. 在UML中, 组成协作图的元素包括(ABD).

- [A],对象
- [B],消息
- [C],发送者
- [D],链

2. 协作图的作用体现在(ABC).

- [A],显示对象及交互关系的空间组织结构
- [B],表现一个类操作的实现
- [C],通过描述对象之间消息的传递情况来反映具体使用语境的逻辑表达
- [D],可以描述对象行为的时间顺序

填空题

1. 在协作图中, 类元角色描述了一个(对象), 关联角色描述了(链), 并通过几何排列表现交互作用中的角色.
2. 交互图是对在一次交互过程中的(对象)和(对象间)的链建模, 显示了对象之间(进行交互)以执行特定用例或用例中特定部分的
3. 在协作图中的链是两个或多个对象之间的(独立连接), 是(关联)的实例.
4. (协作图)通过各个对象之间的组织交互关系以及对象彼此之间的连接, 表达对象之间的交互.
5. 在协作图中, (消息)使用带有标签的箭头表示, 它附在连接发送者和接收者的链上.

简答题

1. 简述使用协作图的原因

描绘对象间消息的传递情况来反映具体的使用语境的逻辑表达, 显示对象及其交互关系的空间组织结构.
表现一个类操作的实现.

2. 描述协作图的元素和他们各自的作用.

对象, 消息和链.
对象: 表示一个或一组对象在完成目标的过程中所应起到的作用.
消息: 描述系统的动态行为.
链: 两个或者多个对象之间的独立连接, 是关联的实例.

3. 简述协作图中消息的种类以及分别在哪种场合使用

同步消息: 用于系统中使用多线程的场合.
异步消息: 用于使用有并行的活动

4. 描述顺序图和协作图的异同

名称	相同点	不同点
顺序图	交互图	表示时间消息序列,没有表示静态对象关系.有效的观察系统的顺序行为
协作图	交互图	表示一个协同中的对象之间的关系和消息以及描述一个操作或分类符的实现.

第十章(活动图)

单选题

1. 活动图中的开始状态使用(C)表示.
- A,菱形

◦ B,直线箭头

◦ C,黑色实心圆

◦ D,空心圆
2. UML中的(B)用来描述过程或操作的工作步骤.
- A,状态图

◦ B,活动图

◦ C,用例图

◦ D,部署图
3. (A)技术是一个活动图中活动状态进行分组，每一组表示一个特定的类,人或部门，他们负责完成组内的活动.
- A,泳道

◦ B,分支

◦ C,分叉汇合

◦ D,转移

多选题

1. 下列说法正确的是(ABD).
- [A],分支将装换路径分成多个部分，每一部分都有单独的监护条件和不同的结果.

◦ [B],一个组合活动在表面上看是一个状态，但其本质却是一组子活动的概括

◦ [C],活动状态是原子性的，用来表示一个具有子结构的纯粹计算的执行

◦ [D],对象流中的对象表示的不仅仅是对象自身，还表示了对象作为过程中的一个状态存在
2. 组成活动图的要素有(ABC).
- [A],泳道

◦ [B],动作状态

◦ [C],活动状态

◦ [D],对象

填空题

1. (活动图)的所有或多数状态都是动作状态或活动状态.
2. (一个对象流)的状态必须与它所表示的参数和结果的类型匹配.
3. (动作状态)的原子性的动作或操作的执行状态，它不能被外部事件的转化中断.
4. 活动状态可以有内部转换，可以有(入口)动作和(出口)动作.

5. 为了组织活动的职责组织而在活动图中将活动状态分为不同的组, 称为(泳道).

简答题

1. 试叙述活动图在软件系统开发中起到的作用.

活动图描述一个操作过程所完成的工, 对用例描述尤其有用.
可建模用例的工作流, 显示用例内部和用例之间的路径.
活动图可以说明用例的实例是如何执行动作以及改变对象状态.
活动图对理解业务处理过程十分有用.

2. 请说明活动图是有哪些元素组成的.

动作状态, 活动状态, 组合状态, 分叉与结合, 分支与合并, 泳道, 对象流.

3. 请简要说明分叉和分支的区别.

分叉用来表示将一个控制流分成两个或者多个并发运行的分支, 分叉具有一个输入转换, 两个或者多个输出转换, 每个转换都可以是独立的控制流.
分支是转换的一部分, 它将转换路径分成多个部分, 每一部分都有单独的监护条件和不同的结果
当动作流遇到分支时, 会根据监护条件(布尔值)的真假来判定动作的流向.
分支的每个路径的监护条件应该是互斥

4. 简要说明活动状态和动作状态的异同点.

动作状态是原子性的动作或操作的执行状态, 它不能被外部事件的转换中断. 动作状态的原子性决定了动作状态要么不执行, 要么就完全执行, 不能中断. 不能有入口动作和出口动作, 也不能有内部转移. 是一种特殊的活动状态. 可以理解为一个原子的活动状态.
活动状态是非原子性的, 用来表示一个具有子结构的纯粹计算的执行. 可以分解成其他子活动或动作状态, 可以被使转换离开状态的事件从外部中断. 活动状态可以有内部转换, 可以有入口动作和出口动作. 至少一个输出完成转换, 当状态中的活动完成时该转换激发.

第十一章(包图)

单选题

1. (A)是用于把元素组织成组的通用机制.
- A,包
 - B,类
 - C,接口
 - D,组件

多选题

1. 包之间的关系总的来讲可以概括为(AB).
- [A],泛化关系
 - [B],依赖关系
 - [C],聚集关系
 - [D],组合关系
2. 下列对于创建包的说法, 正确的是(BCD).
- [A],在序列图和协作图中可以创建包
 - [B],在类图中可以创建包

- [C],如果将包从模型中永久删除, 包及其包中内容都将被删除
- [D],在创建包的依赖关系时, 尽量避免循环依赖

3. 下面(BCD)是构成包图的基本元素.

- [A],发送者
- [B],包
- [C],依赖关系
- [D],子系统

4. 将系统分层很常用的一种方式是将系统分为三层结构, 他们分别是(ABC).

- [A],用户界面层
- [B],数据访问层
- [C],业务逻辑层
- [D],视图层

填空题

1. 组成包图的元素有(包),(子系统)和(依赖关系).
2. 包的可见性关键字包括(public),(private)和(protect).
3. 包是包图中最重要的概念, 它包含了一组(模型元素)和(图).
4. (包图)是一种维护和描述系统总体结构的模型重要建模工具
5. 在UML的建模机制中, (模型)的组织是通过包图来实现的.

简答题

1. 包与包之间的关系.

包和包之间的关系总的来讲可以概括为依赖关系和泛化。

2. 包图的概念和作用

包图是一种维护和描述系统总体结构的模型的重要建模工具, 通过对图中各个包以及包之间关系的描述, 展现出系统的模块与模块之间的依赖关系. 包图是由包和包之间的联系构成的, 它是维护和控制系统总体结构的重要工具。

3. 构成包图的基本元素和主要作用

构成包图的基本元素有包, 子系统和依赖关系。

包: 是一个可以拥有任何种类的模型元素的通用的命名空间。

在系统模型中, 每个图必须被一个唯一确定的包所有, 同样这个包可能被另一个包所包含. 包构成进行配置控制, 存储和访问控制的基础

子系统: 构成了一个大型的复杂系统, 系统和子系统的关系基本上是组合关系。

依赖关系: 通过对包图中各个包以及包之间关系的描述, 展现出系统的模块与模块之间的依赖关系。

4. 请简要阐述包和包图之间的关系.

包是包图中最重要的概念, 是最重要的组成部分. 包图是由包和包之间的联系构成的, 没有包就没有包图。

第十二章(构件图和部署图)

单选题

1. (B)是系统中遵从一组接口且提供实现的一个物理部件，通常指开发和运行时类的物理实现。
- A,部署图
 - B,构件
 - C,类
 - D,接口

多选题

1. 下面的(ABD)元素组成了构件图。
- [A],接口
 - [B],构件
 - [C],发送者
 - [D],依赖关系
2. 部署图的组成元素包括(ABD)。
- [A],处理器
 - [B],设备
 - [C],构件
 - [D],连接
3. 在UML中表示单元的实现是通过(C)和(D).它们描述了系统实现方面的信息，使系统具有可重用性和可操作性。
- [A],包图
 - [B],状态图
 - [C],构件图
 - [D],部署图
4. 在UML中，提供了两种物理表示图形，(A)和(D)。
- [A],构件图
 - [B],对象图
 - [C],类图
 - [D],部署图

填空题

5. 一个构件实例用于表示运行时存在的实现物理单元和在实例节点中的定位，它有两个特征，分别是(代码)和(身份)。
6. 在(构件图)中，将系统中可重用的模块封装称为具有可替代的物理单元，我们称之为构件。
7. 构件图是用来表示系统中(构件)与(构件)之间，以及定义的(类和接口)与构件之间的关系的图。
8. (虚包)是一种只包含从其他包中引入的元素的构件.它被用来提供一个包中某些内容的公共视图。
9. (部署图)描述了一个系统运行时的硬件节点，以及在这些节点上运行的软件构件将在何处物理地运行，以及它们将如何彼此通信的静态视图。

简答题

1. 构建图适用于哪些建模需求

构件图适用于下列建模需求：系统源代码，系统的发布版本，物理数据库，自适应系统，用于建立业务模型和用于系统的开发管理等。

2. 阐述类和构件之间的异同点.

二者都有名称,都可以实现一组接口,都可以参与依赖关系,都可以被嵌套,都可以有实例,都可以参与交互.类描述了软件设计的逻辑组织和意图,而构件这描述了软件设计的物理实现,类可以有属性和操作,构件只有操作,只有通过构件使得接口才能使用.

3. 构件最常见的关系是

构件之间最常见的关系是依赖关系和实现关系.

4. UML主要的三种构件

构件主要包括配置构件,工作产品构件和可执行构件

第十三章(状态图)

单选题

1. 对反应型对象建模一般使用(A).

- A,状态图
- B,顺序图
- C,活动图
- D,类图

多选题

1. 事件可以分为(ABCD).

- [A],信号事件
- [B],改变事件
- [C],调用事件
- [D],时间事件

2. 以下属于组成状态的有(AB).

- [A],顺序
- [B],并发
- [C],同步
- [D],异步

3. 下列对状态图的描述, 正确的是(ACD).

- [A],状态图通过建立类对象的生命周期模型来描述对象随时间变化的动态行为
- [B],状态图适用于描述状态和动作的顺序, 不仅可以展现一个对象拥有的状态, 还可以说明事件如何随着事件的推移来影响这些状态
- [C],状态图的主要目的是描述在对象创建和销毁的过程中资源的不同状态, 有利于开发人员提高开发效率
- [D],状态图描述了一个实体基于事件反应的动态行为, 显示了该实体是如何根据当前所处的状态对不同的事件做反应的

4. 以下构成状态图基本元素(ABC).

- [A],状态

- [B],转化
- [C],初始状态
- [D],链

填空题

1. 在UML中, 状态机由(对象)的各个状态和连接这些状态的转换组成, 是展示状态与状态转换的图
2. (状态图)用于描述模型元素的实例的行为.
3. 状态可以分为(简单状态)和(组成状态).
4. (历史事件)代表上次离开组成状态时的最后一个活动子状态, 它用一个包含字母的小圆圈表示.
5. 在状态机中, 一个(组成事件)的出现可以触发状态的改变.

简答题

1. 状态图由哪些要素构成

由状态, 初始状态, 终止状态, 转换和判定这几个要素构成的

2. 请简要说明状态机和状态图的关系

状态机是一种记录下给定时刻状态的设备, 它可以根据各种不同的输入对每个给定的变化而改变其状态或引发一个动作. 一个状态图(Statechart Diagram)本质上就是一个状态机, 或者是状态机的特殊情况, 它基本上是一个状态机中的元素的一个投影, 这也就意味着状态图包括状态机的所有特征. 状态图描述了一个实体基于事件反应的动态行为, 显示了该实体如何根据当前所处的状态对不同的时间做出反应的.

3. 软件开发中使用状态图的好处

状态图清晰的描述了状态之间的转换顺序, 通过状态的转换顺序可以清晰的看出事件的执行顺序; 清晰的事件顺序有利于程序员在开发程序时避免出现事件错序的情况; 状态图清晰的描述了状态转换时所必须的触发事件, 监护条件和动作等影响转换的因素, 有利于程序员避免程序中非法事件的进入; 状态图通过判定可以更好的描述工作流因为不同的条件发生的分支.

4. 简要回答顺序组成状态和并发组成状态的区别

顺序组成状态的多个子状态之间是互斥的, 不能同时存在的. 一个顺序组成状态最多可以有一个初始状态和一个终态. 并发组成状态可以有两个或者多个并发的子状态, 每个并发子状态还可以进一步分解为顺序组成状态. 一个并发组成状态可以没有初始状态, 终态.