

- ① 阐述面对对象的核心思想？
运用对象、类、继承、聚合、关联、消息和封装等概念和原则来构建软件系统。对象、封装、可重用性和可扩展性。
- ② 阐述面向对象的主要概念与原则，并举例说明。
运用对象、类、继承、聚合、关联、消息和封装等概念和原则。

① 阐述面对对象的核心思想，
对象、封装、可重用性、可扩展性。

② 阐述面向对象的主要概念与原则，并举例说明。
主要概念：运用对象、类、继承、聚合、关联、消息和封装等概念和原则来构建软件系统。

原则：(1) 抽象：指从事物中舍弃个别的、非本质的特征，而抽取共同的、本质特征思维方式。例如可将猫、狗、兔抽象为动物，他们都会叫，生殖但动物这个类并指明是狗叫还是猫叫。
(2) 分类：按照某种原则划分出事物的类别。例猫、狗可分别划分为猫科和犬科。

(3) 封装：一：描述一个事物的性质和行为结合在一起，对外形成该事物的界限。二：信息隐蔽，外界不能直接存取对象的信息内部信息(属性)以及隐藏的内部控制。例使用 Java 创建一个班级类，班级类中有学生人数和按成绩排序的属性和行为。可以将班级中的学生信息设为 private 权限，限制外界直接获取，还可以班级类中加入获取某省份的行为，外界只能调用。

(4) 消息通信：原则上，对象之间只能通过消息进行通信，而不允许对象之外直接访问它内部的属性。例顾客发送购物信息给自助扫码，然后自助扫码发送帐单给顾客。

MEMO NO. 471

(5) 多态性: 一般类和特殊类可以有相同格式的属性或操作, 但这些属性或操作具有不同的含义。例如 Java 的一个函数的传参类型是猫科, 然后让猫科叫一声, 此时传入猫和老虎则会分别出现“喵”和虎啸的行为。

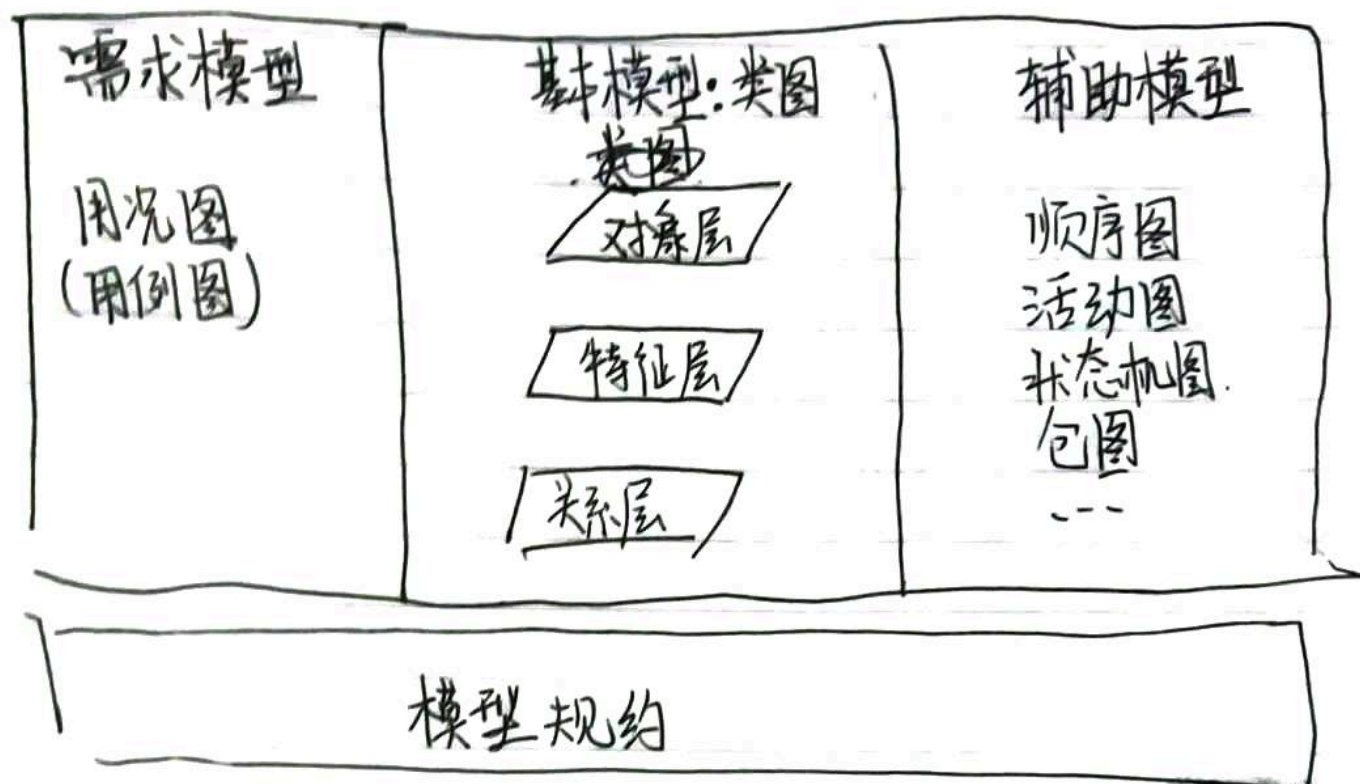
(6) 行为分析。关系机制提供了用关联、继承和聚合等组织类的方法。对系统需要行为分析。例如一个人有儿童、青年、成年、老年的状态, 当一个人^{在系统中时}随着时间推移, 一个人是否可以交养老金, 是否可以结婚等变化。

(7) 复杂性控制: 引入包的概念。例 Java 可以有 bean 包, 专门组成人员类的不同类(学生管理系统)

③ 查阅资料, 讨论 UML 与面向对象方法的关系。

UML 为统一建模语言, 为面向对象开发系统的产品进行说明、可视化和编制文档。UML 中的对象模型、动态模型、功能模型和用例模型共同完成系统的建模, 其中所定义的概念和符号可用于面向对象方法开发用分析、设计和实现的全过程。

① 画图并简述 OOA 模型(各部件组成部分及各种图的作用)及 OOA 过程模型。



需求模型: 使用用况图来捕获与描述用户的要求, 即系统的需求。
基本模型: 用类图构建的模型是系统的静态模型, 它描述系统下用类图构建的模型是系统的基本模型, 因为类图为面向对象编程提供了最直接的依据。基本模型为系统的静态模型, 描述了系统的结构特征。

对象层: 给出系统中所有反映问题域与系统责任的对象。用类符号表达属于一个类的对象的集合。类作为对象的抽象描述是构成系统的基本单位。

特征层: 给出每一个类(及其所代表的对象)的内部特征, 即给出每个类的属性和操作。该层要以分析阶段所能达到的程度为限给出类的内部特征的细节。

关系层: 给出各个类(及其所代表的对象)彼此之间的关系, 包括泛化、关联、依赖。该层描述了对象与外部的联系。

基本模型的三个层次分别描述了 (1) 系统中应设几类对象; (2) 每类对象的内部构成; (3) 每类对象与外部的联系。

辅助模型: 相对基本模型, 系统的行为模型和用包图建立的系统组织模型都作为系统的辅助模型。

模型规约: 对各种图中的建模元素按一定要求进行规约。

用例图: 用于对系统的功能以及与系统进行交互的外部事物建模。找到系统的参与者, 描述参与者如何使用系统以及系统提供的功能。

类图: 用于描述系统中所包含的类以及它们之间的关系, 帮助简化系统, 它是系统分析和设计的重要产物也是系统编码和测试的重要依据。

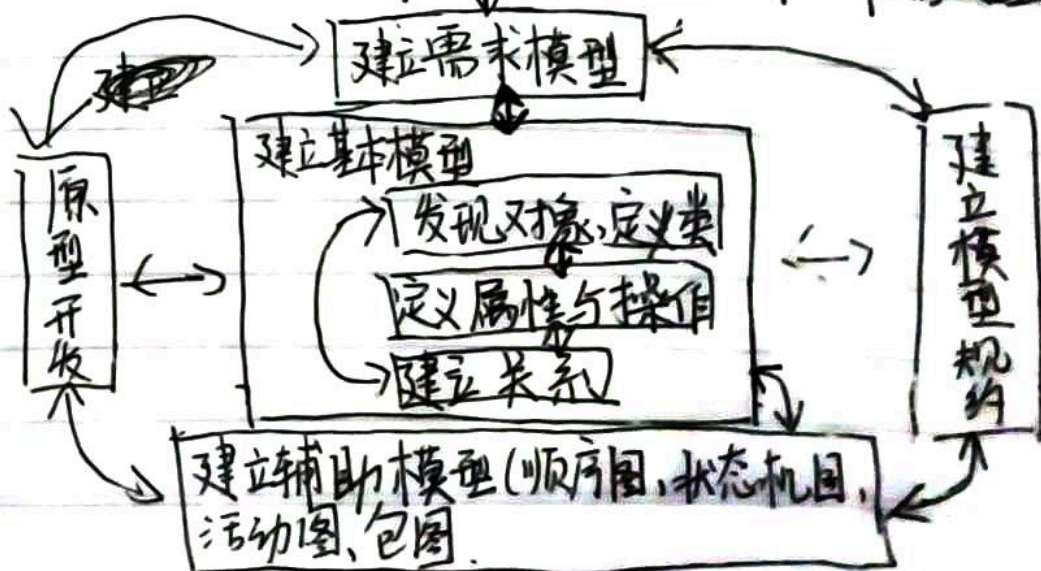
顺序图: 强调消息时间顺序的交互图, 描述对象之间传送消息的时间顺序, 用于表示用例中行为的顺序。

活动图: 描述活动的顺序, 展现从一个活动到另一个活动的控制流。

状态机图: 说明对象在其生命周期中的状态变化, 这些变化由响应事件或监控条件触发。

包图: 由包和包之间的关系构成, 维护和控制系统的总体结构。

OA 过程模型:



阐述UML中图的分类,以及各类图的作用。

- ① 用例图: 用于描述系统功能的动态视图。由参与者、用例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。作用: (1) 表示系统需求的分析, (2) 作为开发过程的基础, 驱动各阶段的作用。 (3) 用于验证和确认系统需求。
- ② 类图与对象图: 用于描述系统的静态结构。类图显示了系统的静态结构, 而系统的静态结构构成了系统的概念基础。一个类图通过系统中的类以及各个类之间的关系来描述系统的静态方面。作用: (1) 为系统的词汇建模 (2) 模型化简单的写作 (3) 模型化逻辑数据库模式。
- ③ 序列图: 用于表现一个交互, 该交互是一个协作中各类元角色的一组信息交换, 侧重于强调时间顺序, 是对象之间基于时间顺序的动态交互, 显示出随着时间的变化对象之间是如何进行通信的。作用: 更加详细描述用例表达的需求。
- ④ 协作图: 着重于对系统成分如何协同工作进行描述。协作图包含一组对象和以消息交互为练习的关联, 用于描述系统的行为是如何由系统的成分合作实现的。
- ⑤ 活动图: 用于描述系统行为的模型视图, 它可用来描述动作和动作导致对象状态改变的结果, 而不用考虑引发状态改变的事件。
- ⑥ 包图: 用于维护和描述系统总体结构模型, 通过各包之间的描述, 展现出系统的模块与模块之间的依赖关系。包图用于对系统进行组织, 反映系统结构。

⑦ 构件图：用来表示系统中构件与构件之间，以及定义的类或接口与构件之间关系的图。

⑧ 状态图：用来表示系统的行为，清晰描述状态之间的转换顺序，描述状态转换时所必须触发事件、监护条件和动作影响转换的因素。