

cpu:

1、运算器：

- 1) 算术逻辑单元 ALU ( Arithmetic & Logic Unit )
- 2) 累加寄存器 AC ( Accumulator )
- 3) 数据缓冲寄存器 DR ( Data Register )
- 4) 状态条件寄存器 PSW ( Program Status Word )

2、控制器：

- 1) 指令寄存器 IR ( Instruction Register )
- 2) 程序计数器 PC ( Program Counter )
- 3) 地址寄存器 AR ( Address Register )
- 4) 指令译码器 ID ( Instruction Decoder )

3、寄存器组

- 1) 专用寄存器 ( 运算器、控制器中的寄存器，作用固定 )
- 2) 通用寄存器 ( 程序员规定用途 )

4、内部总线

## 上午题型

# 下午题型

## 第 1 道 数据流图 DFD ( 必答 )

### 考点

- 补充数据流图外部实体
- 补充数据流图数据存储
- 补充数据流 ( 名称、起点、终点 )
- 数据流图的改错 ( 较少考察 , 包括数据流错误、删除多余数据流 )
- 数据流图相关概念简答

### 笔记

1. 章节 : 第 6 章第 2 节
2. 页码 : 325-336
3. 简介 :

## 第 2 道 数据库设计 ( 必答 )

### 考点

- 补充 E-R 图
- E-R 图转化为关系模式
- 主键和外键、新增联系判断

## 笔记

1. 章节：第 9 章第 2 节
2. 页码：466-473
3. 简介：

## 第 3 道 UML 建模（必答）

### 考点

- 用例图（联系类型，参与者）
- 类图和对象图（多重度，联系类型）
- 顺序图（补充对象名和消息名）
- 活动图（补充活动名，分岔线用途）
- 状态图（补充状态，状态转换条件）
- 通信图（补充对象名，消息名）

## 笔记

1. 章节：第 7 章第 2 节
2. 页码：363—377（pdf：375-389）
3. 简介：**UML（Unified Modeling Language，统一建模语言）是面向对象软件的标准化建模语言。**
4. 要素：**3 个**。UML 的基本构造块、支配这些构造块如何放置在一起的规则和运用与整个语言的一些公共机制。

5. 构造块：3 种。事物、关系和图。
6. 事物：4 种。事物是对模型中最具有表达性的成分的抽象。结构事物、行为事物、分组事物和注释事物。
7. 结构事物 ( Structural Thing )：包括类 ( Class )、接口 ( Interface )、协作 ( Collaboration )、用例 ( Use Case )、主动类 ( Active Class )、构件 ( Component )、制品 ( Artifact ) 和结点 ( Node )。

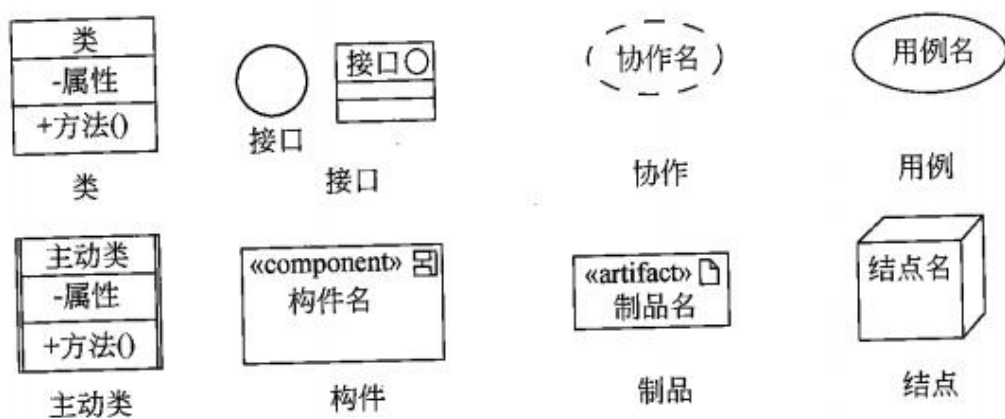


图 7-2 结构事物的图形表示

8. 行为事物 ( Behavior Thing )：包括交互 ( Interaction )、状态机 ( State Machine ) 和活动 ( Activity )。



图 7-3 行为事务的图形表示

9. 分组事物：最主要的分组事物是包。

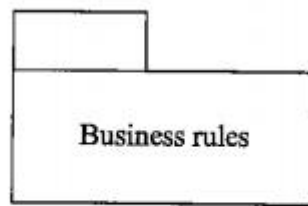


图 7-4 包

10. 注释事物：注解是一种主要的注释事物。

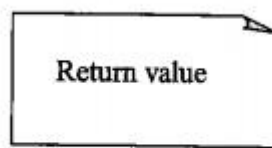


图 7-5 注解

11. 关系：4 种。关系把事物结合在一起。依赖、关联、泛化和实现。



图 7-6 依赖



图 7-7 关联



图 7-8 聚集



图 7-9 泛化



图 7-10 实现

12. 图：13 种。图聚集了相关的事物。类图、对象图、用例图、序列图（或顺序图、时序图）、通信图（或协作图）、状态图、活动图、构件图、组合结构图、部署图、包图、交互概览图和计时图。序列图、通信图、交互概览图和计时图均被称为交互图。

13. 类图 ( Class Diagram )：类图展现了一组对象、接口、协作和它们之间

的关系。包括类、接口、协作、关系（依赖、泛化和关联）。

14. 对象图 ( Object Diagram ) : 对象图展现了某一时刻一组对象以及它们之间的关系，描述了在类图中所建立的事物的实例的静态快照。

15. 用例图( Use Case Diagram ) :用例图展现了一组用例、参与者( Actor )以及它们之间的关系。包括用例、参与者、用例之间的扩展关系 ( <<extend>> ) 和包含关系 ( <<include>> ), 参与者和用例之间的关联关系，用例与用例以及参与者与参与者之间的泛化关系。

16. 交互图 之 序列图( Sequence Diagram ) :序列图是场景的图形化表示，描述了以时间顺序组织的对象之间的交互活动。序列图有 2 个不同于通信图的特征：序列图有对象生命线、序列图有控制焦点。序列图和通信图是同构的，可以互相转换。

17. 交互图 之 通信图( Communication Diagram ) :通信图强调收发消息的对象的结构组织。通信图也称作协作图。通信图有 2 个不同于序列图的特性：通信图有路径、通信图有顺序号。序列图和通信图是同构的，可以互相转换。

18. 交互图 之 交互概览图 ( Interaction Overview Diagram ) : 描述业务过程中的控制流概览，软件过程中的详细逻辑概览，以及将多个图进行连接，抽象掉了消息和生命线。交互概览图是活动图的变体。

19. 交互图 之 计时图 ( Timing Diagram ) : 描述对象状态随着时间改变的情况。

20. 状态图 ( State Diagram ) : 展现了一个状态机。状态图由状态、转换、事件和活动组成。

21. 活动图 ( Activity Diagram ) : 展现了在系统内从一个活动到另一个活动的流程。活动图是一种特殊的状态图。包括活动状态和动作状态、转换和对象。活动图可以表示分支、合并、分岔和汇合。
22. 构件图 ( Component Diagram ) : 展现了一组构件之间的组织和依赖。
23. 组合结构图 ( Composite Structure Diagram ) : 用于描述一个分类器 ( 如类、构建或用例 ) 的内部结构 , 分类器与系统中其他组成部分之间的交互端口 , 展示一组相互协作的实例如何完成特定的任务 , 描述设计、架构模式或策略。
24. 部署图 ( Deployment Diagram ) : 用来对面向对象系统的物理方面建模的方法。与构件图相关。
25. 包图 ( Package Diagram ) : 用于把模型本身组织称层次结构的通用机制 , 不能执行 , 展现由模型本身分解而成的组织单元以及其间的依赖关系。

## 第 4 道 C 算法设计 ( 必答 )

### 考点

- 各种经典算法设计和数据结构 , 如链表、栈、二叉树操作算法、KMP 算法等
- 算法类型 ( 动态规划法、分治法、回溯法、递归法、贪心法 )
- 时间复杂度、空间复杂度
- 给定输入求输出

## 笔记

1. 章节：第 8 章、第 3 章
2. 页码：416-454、99-179
3. 简介：

## 第 5 道 C++ 语言程序设计（选答）

### 考点

- 简单语法
- 设计模式

## 笔记

## 第 6 道 Java 语言程序设计（选答）

### 考点

- 简单语法
- 设计模式

## 笔记