



扫码小程序获取更多【软考备考资料】

第1章 计算机组成与体系结构

1 考情分析

根据对历年的考试真题进行分析，本章要求考生掌握以下几个方面的知识：

- (1) 掌握数据的进制转换（十进制、二进制、十六进制互转）。
- (2) 掌握原码/反码/补码/移码的转换规则，了解各种码制的表示范围和表示个数（注意补码和移码的人为定义）。
- (3) 掌握浮点数的格式，及各部分含义，了解浮点数运算的逻辑过程。
- (4) 掌握 CPU 各子部件的功能及其分类。
- (5) 了解 Flynn 分类法的分类及各类的典型系统。
- (6) 了解指令的基本分类、寻址方式分类。
- (7) 了解 CISC 与 RISC 的特点区别。
- (8) 掌握流水线相关的参数计算（流水线周期、流水线建立时间、流水线执行时间、流水线执行时间实践公、流水线吞吐率、流水线最大吞吐率等）。
- (9) 了解计算机分层存储结构相关概念（效率、容量、成本对比，局部性原理支撑）。
- (10) 了解局部性原理，熟悉 Cache 的相关概念和特点，区分 Cache 三种映射方式。
- (11) 掌握主存计算题型，掌握相关参数的计算（存储单元个数，存储总容量，单位芯片容量，芯片个数）。
- (12) 了解总线的分类-数据总线、控制总线、地址总线。
- (13) 掌握系统可靠性计算（串联系统、并联系统、混联系统）。
- (14) 了解奇偶校验，了解 CRC 校验（利用模二除计算），掌握海明校验位计算，区分三种校验方式对检错和纠错的特点。
- (15) 了解常见的计算机性能指标。

1.1 本章重点

序号	知识领域	知识点详情
1	数据的表示 (★★★★)	进制的转换
2		原码/反码/补码/移码表示
3		浮点数的表示
1	运算器与控制器 (★★★★)	运算器与控制器
1	Flynn 分类法 (★★)	Flynn 分类法
1	CISC 与 RISC (★★)	CISC 与 RISC
1	流水线技术 (★★★★)	流水线技术
1	存储系统 (★★★★)	分级存储体系
2		Cache
3		内存
1	总线系统 (★)	总线系统

1	可靠性 (★)	可靠性
1	校验码 (★★★★)	校验码
1	计算机性能指标 (★)	性能指标

2 考点精讲

2.1 数据的表示 (★★★★)

2.1.1 进制的转换

【考法分析】

本考点的基本考法是与内存地址计算、IP 地址计算结合考查。

【要点分析】

- 1、十进制转 R 进制 (短除法);
- 2、R 进制转十进制 (按权展开法);
- 3、二进制转八进制、转十六进制 (分组快速转换)。

【备考点拨】

- 1、掌握二进制数与十进制数的互转，熟练计算 IP 地址转换;
- 2、掌握二进制与十六进制的互转，熟练掌握内存地址转换。

2.1.2 原码/反码/补码/移码表示

【考法分析】

本考点的主要考查方式有：给定一些描述，让考生判断是否正确；计算某种码制的表示范围或表示数值的个数；不同码制的表示。

【要点分析】

- 1、原码/反码/补码/移码的转换规则：

原码：符号位+数值位绝对值。

反码：正数的反码是原码本身，负数的反码是符号位不变数值位按位取反。

补码：正数的补码是原码本身，负数的补码是符号位不变数值位在反码基础上加 1。

移码：正数和负数的移码都是在补码基础上符号位取反。

- 2、原码/反码/补码/移码的表示范围 (如下图所示)：

码制	定点整数	定点小数	表示数码个数
原码	$-(2^{n-1}-1) \sim +(2^{n-1}-1)$	$-(1-2^{-(n-1)}) \sim +(1-2^{-(n-1)})$	2^n-1
反码	$-(2^{n-1}-1) \sim +(2^{n-1}-1)$	$-(1-2^{-(n-1)}) \sim +(1-2^{-(n-1)})$	2^n-1
补码	$-2^{n-1} \sim +(2^{n-1}-1)$	$-1 \sim +(1-2^{-(n-1)})$	2^n
移码	$-2^{n-1} \sim +(2^{n-1}-1)$	$-1 \sim +(1-2^{-(n-1)})$	2^n

【定义-0 的补码为其 -2^{n-1} 的补码，以 $n=8$ 为例，人为定义 10000000 是 -128 的补码】

- 3、对于 0 的特殊化：0 在原码和反码中有 +0 和 -0 两种表示方式，而对于补码和移码其表示方式一致。

【备考点拨】

- 1、掌握原码/反码/补码/移码的转换；
- 2、掌握原码/反码/补码/移码的表示范围和能够表示数的个数，也可能与浮点数表示结合考查浮点数的表示范围；
- 3、注意在补码和移码中 -0 的特殊性和人为定义。

2.1.3 浮点数的表示

【考法分析】

本知识点的考查方式为：给定一些描述（关于浮点数构成、浮点数运算规则等）让学员判断是否正确；判断尾数和解码对浮点数的影响。

【要点分析】

- 1、浮点数各部分表示意义： $N = \text{尾数} \times \text{基数}^{\text{指数}}$
 - (1) 一般尾数用补码，阶码用移码；
 - (2) 阶码的位数决定数的表示范围，位数越多范围越大；
 - (3) 尾数的位数决定数的有效精度，位数越多精度越高。
- 2、浮点数运算规则：对阶 > 尾数计算 > 结果格式化
 - (1) 对阶时，小数向大数看齐；
 - (2) 对阶是通过较小数的尾数右移实现的。

【备考点拨】

- 1、掌握浮点数各部分的意义，重点掌握阶码、尾数的意义；
- 2、掌握浮点数运算的过程和简单规则。

2.1.4 逻辑运算

【考法分析】

本知识点的考查形式为：给定一些运算表达式让学员计算结果或找到等价表达式；判断短路运算。

【要点分析】

1、运算符：

较高优先级（关系运算符）：< (小于)；<= (小于或等于)；> (大于)；>= (大于或等于)；

较低优先级（关系运算符）：== 等于；!= 不等于

逻辑运算符：

逻辑或（||、+、∪、∨、OR）：连接的两个逻辑值全 0 时才取 0

逻辑与（&&、*、·、∩、∧、AND）：连接的两个逻辑值全 1 时才取 1

逻辑异或（⊕、XOR）：连接的两个逻辑值不相同才取 1，相同则取 0

逻辑非（!、¬、~、NOT、—）：将原逻辑值取反即可

2、优先级

！（非）->&&（与）->||（或）

逻辑运算符中的“&&”和“||”低于关系运算符，“!”高于算术运算符

因此运算符的优先顺序为：! > 算术运算符 > 关系运算符 > && > || > 赋值运算符

3、短路原则：在逻辑表达式的求解中，并不是所有的逻辑运算符都要被执行。

(1) $a \&\&b \&\&c$ 只有 a 为真时，才需要判断 b 的值，只有 a 和 b 都为真时，才需要判断 c 的值。(2) $a || b || c$ 只要 a 为真，就不必判断 b 和 c 的值，只有 a 为假，才判断 b。a 和 b 都为假才判断 c。

【备考点拨】

- 1、掌握运算符写法和计算规则；
- 2、掌握短路原则。

2.2 运算器与控制器 (★★★★)

【考法分析】

本考点主要的考查形式有：给定部件判断部件所属运算器或控制器；给定一定功能描述，判断具体描述的部件。

【要点分析】

1、运算器：

- ① 算术逻辑单元 ALU：数据的算术运算和逻辑运算
- ② 累加寄存器 AC：通用寄存器，为 ALU 提供一个工作区，用在暂存数据
- ③ 数据缓冲寄存器 DR：写内存时，暂存指令或数据
- ④ 状态条件寄存器 PSW：存状态标志与控制标志（争议：也有将其归为控制器的）

2、控制器：

- ①程序计数器 PC：存储下一条要执行指令的地址
- ②指令寄存器 IR：存储即将执行的指令
- ③指令译码器 ID：对指令中的操作码字段进行分析解释
- ④时序部件：提供时序控制信号

【备考点拨】

- 1、掌握运算器、控制器分类所属的部件，并熟悉各个部件的功能和特点能够加以区分。

2.3 Flynn 分类法 (★★)

【考法分析】

本题考查形式主要是根据题干描述判断所属分类或判断各个分类的典型系统。

【要点分析】

体系结构类型	结构	关键特性	代表
单指令流单数据流 SISD	控制部分：一个 处 理 器：一个 主存模块：一个		单处理器系统
单指令流多数据流 SIMD	控制部分：一个 处 理 器：多个 主存模块：多个	各处理器以异步的形式执行同一条指令	并行处理机 阵列处理机 超级向量处理机
多指令流单数据流 MISD	控制部分：多个 处 理 器：一个 主存模块：多个	被证明不可能，至少是不实际	目前没有，有文献称流水线计算机为此类
多指令流多数据流 MIMD	控制部分：多个 处 理 器：多个 主存模块：多个	能够实现作业、任务、指令等各级全面并行	多处理机系统 多计算机

【备考点拨】

- 1、掌握 Flynn 分类法四种类型的特点及其相关典型系统
- 2、常考典型系统：阵列处理机、超级向量处理机；常考类型：SIMD、MISD

2.4 CISC 与 RISC (★★)

【考法分析】

本考点的基本考法是给出一些 CISC 与 RISC 的特点，让考生判断说法的正确性；给出一定描述判断寻址方式。

【要点分析】

- 1、CISC：复杂指令集；RISC：精简指令集。
- 2、CISC 的特点：指令数量多，指令频率差别大，多寻址，使用微码。
- 3、RISC 的特点：指令数量少，操作寄存器，单周期，少寻址，多通用寄存器，硬布线逻辑控制，适用于流水线。

指令系统类型	指令	寻址方式	实现方式	其它
CISC (复杂)	数量多，使用频率差别大，可变长格式	支持多种	微程序控制技术 (微码)	研制周期长
RISC (精简)	数量少，使用频率接近，定长格式，大部分为单周期指令，操作寄存器，只有 Load/Store 操作内存	支持方式少	增加了通用寄存器；硬布线逻辑控制为主；适合采用流水线	优化编译，有效支持高级语言

4、寻址方式

立即寻址方式

特点：操作数直接在指令中，速度快，灵活性差

直接寻址方式

特点：指令中存放的是操作数的地址

间接寻址方式

特点：指令中存放了一个地址，这个地址对应的内容是操作数的地址。

寄存器寻址方式

特点：寄存器存放操作数

寄存器间接寻址方式

特点：寄存器内存放的是操作数的地址

【备考点拨】

1、熟记 CISC 与 RISC 特点差异。

2、识别寻址方式。

2.5 流水线技术 (★★★★)

【考法分析】

本考点涉及的考查形式有：

- (1) 流水线相关理论概念；
- (2) 流水线相关计算。

【要点分析】

流水线相关概念：

(1) 流水线是指在程序执行时多条指令重叠进行操作的一种准并行处理实现技术。各种部件同时处理是针对不同指令而言的，它们可同时为多条指令的不同部分进行工作，以提高各部件的利用率和指令的平均执行速度。

(2) 流水线建立时间：1 条指令执行时间。

(3) 流水线周期：执行时间最长的一段，记作 t 。

2、流水线相关计算：

(1) 流水线执行时间（理论公式）： $(t_1+t_2+\cdots+t_k)+(n-1)*t$

(2) 流水线执行时间（实践公式）： $k*t + (n-1)*t$

(3) 流水线吞吐率：

$$TP = \frac{\text{指令条数}}{\text{流水线执行时间}}$$

(4) 流水线最大吞吐率

$$TP_{\max} = \lim_{n \rightarrow \infty} \frac{n}{(k+n-1)t} = \frac{1}{t}$$

(5) 流水线加速比：

$$S = \frac{\text{不使用流水线执行时间}}{\text{使用流水线执行时间}}$$

【备考点拨】

1、了解流水线相关理论概念；

2、掌握流水线相关计算公式。

2.6 存储系统 (★★★★)

2.6.1 分级存储体系

【考法分析】

本考点主要考查的形式一般为判断相关概念描述的正确性，考查较为频繁的概念有局部性原理、各层次的速度比较、Cache 的位置；识别部分存储器。

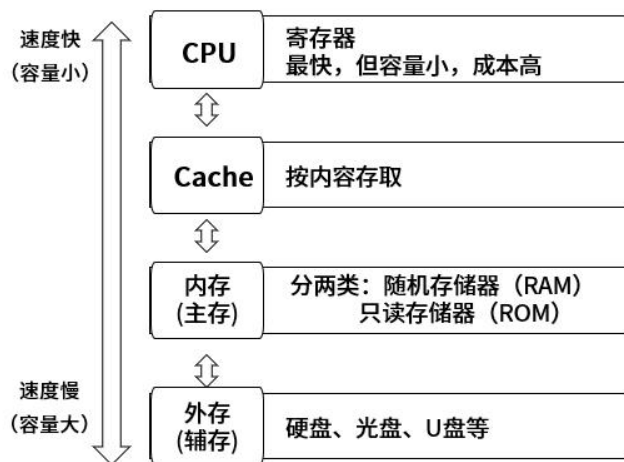
【要点分析】

- (1) 关于多级存储体系的分布（见下图）；
- (2) 关于多级存储体系不同层次的大小、速度等对比（见下图）；
- (3) 关于多级存储体系的理论支持体系（局部性原理）。

时间局部性：指程序中的某条指令一旦执行，不久以后该指令可能再次执行，典型原因是由于程序中

存在着大量的循环操作。

空间局部性：指一旦程序访问了某个存储单元，不久以后，其附近的存储单元也将被访问，即程序在一段时间内所访问的地址可能集中在一定的范围内，其典型情况是程序顺序执行。



局部性原理是层次化存储结构的支撑

(4) 存储器分类

1、存储器位置

内存&外存

2、存取方式

(1) 按内容存取：

相联存储器（如 Cache）

(2) 按地址存取：

随机存取存储器（如内存）

顺序存取存储器（如磁带）

直接存取存储器（如磁盘）

3、工作方式

(1) 随机存取存储器 RAM（如内存 DRAM）

(2) 只读存储器 ROM（如 BIOS）

【备考点拨】

1、掌握相关理论概念。

2、了解典型存储部件对应的存储器类型。

2.6.2 Cache

【考法分析】

一般以选择题形式考查，要求判断相关概念描述是否正确，Cache 映射方式的对比区分，Cache 映射方式的控制由硬件直接控制。

【要点分析】

1、Cache 的相关概念：理论依据（局部性原理），大小、速度、成本等对比。

在计算机的存储系统体系中，Cache 是（除寄存器以外）访问速度最快的层次。

Cache 改善了系统性能，提高了 CPU 访问内存的效率。

Cache 的地址映像由硬件直接完成。（常考）

2、Cache 映射方式：全相联、组相联、直接相联映像的对比（冲突率依次增高，电路复杂度依次降低）：

直接相联映像：硬件电路较简单，但冲突率很高。Cache 对应页号位置已有数据即冲突需淘汰页面再调入。

全相联映像：电路难于设计和实现，只适用于小容量的 cache，冲突率较低。所有位置已有数据才冲突需要淘汰页面后再调入。

组相联映像：直接相联与全相联的折中。

	冲突率 (高、中、低)	电路复杂度 (复杂、简单、折中)	其他
直接相联映像	高	简单	对应位置有数据即冲突
全相联映像	低	复杂	所有位置有数据即冲突
组相联映像	中	折中	

【备考点拨】

- 1、掌握 Cache 相关概念；
- 2、掌握 Cache 相联映像方式的对比。

2.6.4 内存

【考法分析】

本知识点的考查主要是对相关参数的计算：内存总容量、芯片单位容量、芯片片数。总片数=总容量/每片的容量。

【要点分析】

- 1、内存单元数计算：最大地址+1-最小地址
- 2、内存总容量：按字节编址，内存单元数*8bit；按字编址，内存单元数*机器字长。
- 3、已知芯片单位容量，求所用芯片的片数，总容量/单位容量；
- 4、已知所用芯片的片数，求取芯片单位容量，总容量/芯片片数。

【备考点拨】

1、掌握相关计算，注意结合二进制与十六进制、十进制之间的相互转化，注意编址方式的选择会影响内存总容量。

2.7 总线系统 (★)

【考法分析】

本知识点考查形式主要是判断总线的分类。

【要点分析】

- 1、数据总线 (Data Bus)：在 CPU 与 RAM 之间来回传送需要处理或是需要储存的数据。
- 2、地址总线 (Address Bus)：用来指定在 RAM (Random Access Memory) 之中储存的数据的地址。
- 3、控制总线 (Control Bus)：将微处理器控制单元 (Control Unit) 的信号，传送到周边设备。

【备考点拨】

1、掌握总线的分类，能够区分数据总线、控制总线、地址总线的区别。

2.8 可靠性 (★)

【考法分析】

本知识点主要考察关于不同系统类型的可靠性计算；识别可靠性/可用性、可维护性指标表示。

【要点分析】

- 1、串联系统计算： $R_{\text{总}} = R_1 * R_2 * \dots * R_n$ ；
- 2、并联系统计算： $R_{\text{总}} = 1 - (1 - R_1) (1 - R_2) \dots (1 - R_n)$ ；
- 3、N 模混联系统：先将整个系统划分为多个部分串联 R_1 、 $R_2 \dots$ 等，再计算 R_1 、 R_2 内部的并联可靠性，带入原公式。
- 4、可靠性指标



平均无故障时间 \rightarrow (MTTF) $MTTF=1/\lambda$, λ 为失效率
 平均故障修复时间 \rightarrow (MTTR) $MTTR=1/\mu$, μ 为修复率
 平均故障间隔时间 \rightarrow (MTBF) $MTBF = MTTR + MTTF$
 系统可用性 $\rightarrow MTTF/(MTTR+MTTF) \times 100\%$
 在实际应用中, 一般 MTTR 很小, 所以通常认为 $MTBF \approx MTTF$ 。
 可靠性可以用 $MTTF/(1+MTTF)$ 来度量。
 可维护性可以用 $1/(1+MTTR)$ 来度量

【备考点拨】

- 1、掌握可靠性的计算。
- 2、了解可靠性/可用性、可维护性的指标表示。

2.9 校验码 (★★★)

【考法分析】

本考点主要考查形式有: 给定校验码相关基本概念, 判断正误; 区分不同校验方式的特点; 对于海明校验码的计算问题, 包括校验码的位数计算, 校验码的位置计算。

【要点分析】

- 1、奇偶校验: 掌握校验原则等相关概念, 只检奇数位错, 不能纠错。
- 2、循环校验码 CRC: 可查错, 不可纠错, 运用模二除法计算校验码。
- 3、海明校验: 要求掌握相关概念, 可查错, 可纠错; 要求掌握海明校验码校验位计算: $2^r \geq r+m+1$ 。
- 4、校验码对比

	校验码位数	校验码位置	检错	纠错	校验方式
奇偶校验	1	一般拼接在头部	可检奇数位错	不可纠错	奇校验: 最终 1 的个数是奇数个; 偶校验: 最终 1 的个数是偶数个;
CRC 循环冗余校验	生成多项式最高次幂决定	拼接在信息位尾部	可检错	不可纠错	模二除法求余数, 拼接作为校验位
海明校验	$2^r \geq m+r+1$	插入在信息位中间	可检错	可纠错	分组奇偶校验

【备考点拨】

- 1、掌握奇偶校验的规则, 掌握其特性;
- 2、了解 CRC 校验的规则, 了解模二除法, 掌握其特性;
- 3、掌握海明校验的规则, 重点掌握其校验位计算公式 $2^r \geq r+m+1$ 。了解其编码过程。

2.10 计算机性能指标 (★)

【考法分析】

本考点主要考查形式有: 给出描述, 判断对应的性能指标; 给出一定参数, 计算性能指标的参数。

【要点分析】

- 1、主频 (计算机参数), 时钟周期=主频的倒数, 主频=倍频*外频。
- 2、平均每条指令的平均时钟周期个数 (CPI, clock per instruction), $CPI=\text{时钟周期总数}/\text{指令总条数}$ 。
- 3、每 (时钟) 周期运行指令条数 (IPC, instruction per clock), $IPC=\text{指令总条数}/\text{时钟周期总数}$
- 4、百万条指令每秒 (MIPS, Million Instructions Per Second) $= (IPC \times \text{时钟周期}) / 10^6$
- 5、每秒百万个浮点操作 (MFLOPS, Million Floating-point Operations per Second), 与 MIPS 相似, 针对的是浮点操作。
- 6、字长 (计算机参数)。

- 7、**总线宽度**：每次脉冲通过的数据量。
- 8、**带宽**：单位时间通过的数据量，带宽=数据总量/总时间
- 9、**吞吐量**，某个时间段内完成的任务总数；吞吐率，单位时间内完成的任务总数，吞吐率=任务总数/总时间。

【备考点拨】

- 1、了解常见计算机性能指标的描述和意义；
- 2、了解常见参数的计算。

3 章节问答

1、浮点数的规格化需要掌握吗？

答：

浮点数只需要掌握运算的逻辑过程有规格化，具体的规格化过程在软设中没有涉及。

2、状态条件寄存器 PSW 属于运算器还是控制器？

答：

PSW 在运算过程中会保存为 0 标识、溢出标识等，而这些标识在某些过程中会用来作为控制跳转的条件，所以在分类过程中，PSW 既可以归于运算器，也可以归于控制器，存在争议。建议做题时如果碰到分类问题，将 PSW 放到最后考虑。

3、流水线执行时间计算有理论公式和实践公式，应该选择哪一个进行计算呢？

答：

计算流水线执行时间过程中，默认选择理论公式，在做题中，如果理论公式没有正确答案，才会考虑用实践公式计算。

4、Cache 的地址映射是程序员通过应用程序控制的吗？

答：

Cache 对于程序员来说是不了解、不会使用的存储层次，也就是说 Cache 对于程序员来说是透明的。Cache 与内存之前的映射是由硬件直接完成的。

第 2 章 操作系统

1 考情分析

根据对历年的考试真题进行分析，本章要求考生掌握以下几个方面的知识：

- (1) 了解进程状态变迁。
- (2) 掌握 PV 操作相关解题技巧。
- (3) 了解死锁的条件，掌握死锁资源分配计算、银行家算法。
- (4) 了解页式存储、段式存储、段页式存储的特点，掌握页式存储页表的使用以及逻辑地址与物理地址的转换，掌握段式存储合法段地址的判断。
- (5) 掌握磁盘相关解题技巧。（磁盘存取时间计算，磁盘优化存储问题，移臂调度算法问题）
- (6) 掌握索引文件相关计算。
- (7) 掌握树形目录绝对路径与相对路径区分，了解文件名写法。
- (8) 掌握位示图相关计算。
- (9) 区分 I/O 控制方式（程序控制（查询）方式、程序中断、DMA）特点。

1.1 本章重点

序号	知识领域		知识点详情
1	进程管理	进程的状态 (★★)	进程的状态
2		前趋图 (★★★)	前趋图
3		信号量与 PV 操作 (★★★★)	信号量与 PV 操作
4		死锁及银行家算法 (★★★★)	死锁及银行家算法
1	存储管理	段页式存储 (★★★★)	页式存储
2			段式存储
3			段页式存储
4		页面置换算法 (★)	页面置换算法
5		磁盘管理 (★★)	磁盘管理

1	文件管理	绝对路径与相对路径 (★★★)	绝对路径与相对路径
1		索引文件 (★★)	索引文件
2		位示图 (★★)	位示图
1	设备管理	I/O 设备管理 (★)	I/O 设备管理

2 考点精讲

【操作系统概述】

2.1 操作系统相关概念

【考法分析】

本考点主要考查形式是识别操作系统层次、判断操作系统相关描述是否正确、判断线程部分内容是否能够共享。

【要点分析】

1、软件分层：



2、操作系统功能：

管理系统的硬件、软件、数据资源
控制程序运行
人机之间的接口
应用软件与硬件之间的接口

3、特殊的操作系统

分类	特点
批处理操作系统	单道批：一次一个作业入内存，作业由程序、数据、作业说明书组成 多道批：一次多个作业入内存，特点：多道、宏观上并行微观上串行
分时操作系统	采用时间片轮转的方式为多个用户提供服务，每个用户感觉独占系统 特点：多路性、独立性、交互性和及时性
实时操作系统	实时控制系统和实时信息系统 交互能力要求不高，可靠性要求高（规定时间内响应并处理）
网络操作系统	方便有效共享网络资源，提供服务软件和有关协议的集合 主要的网络操作系统有：Unix、Linux 和 Windows Server 系统
分布式操作系统	任意两台计算机可以通过通信交换信息 是网络操作系统的更高级形式，具有透明性、可靠性和高性能等特性
微机操作系统	Windows: Microsoft 开发的图形用户界面、多任务、多线程操作系统 Linux: 免费使用和自由传播的类 Unix 操作系统，多用户、多任务、多线程和多 CPU 的操作系统
嵌入式操作系统	运行在智能芯片环境中 特点：微型化、可定制（针对硬件变化配置）、实时性、可靠性、易移植性（HAL 和 BSP 支持）

4、嵌入式操作系统特点

微型化。从性能和成本角度考虑，希望占用资源和系统代码量少，如内存少、字长短、运行速度有限、能源少（用微型电池）。

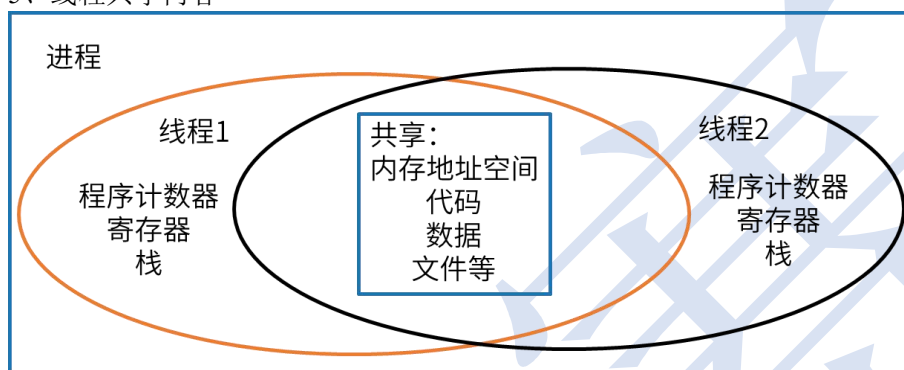
可定制。从减少成本和缩短研发周期考虑，要求嵌入式操作系统能运行在不同的微处理器平台上，能针对硬件变化进行结构与功能上的配置，以满足不同应用需要。

实时性。嵌入式操作系统主要应用于过程控制、数据采集、传输通信、多媒体信息及关键要害领域需要迅速响应的场合，所以对实时性要求高。

可靠性。系统构件、模块和体系结构必须达到应有的可靠性，对关键要害应用还要提供容错和防故障措施。

易移植性。为了提高系统的易移植性，通常采用硬件抽象层 (Hardware Abstraction Level; HAL) 和板级支持包 (Board Support Package, BSP) 的底层设计技术。

5、线程共享内容



【备考点拨】

- 1、了解操作系统的分层和任务。
- 2、了解特殊的操作系统分类和相应的一些特点。
- 3、了解线程能够共享的部分内容。

【进程管理】

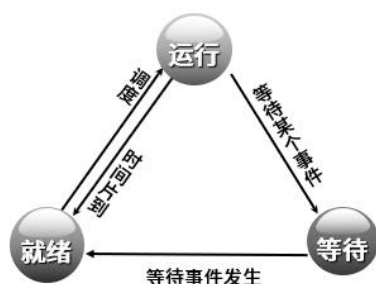
2.1 进程的状态 (★★)

【考法分析】

本考点主要考查形式主要是根据图示判断相关状态位置或状态变迁条件。

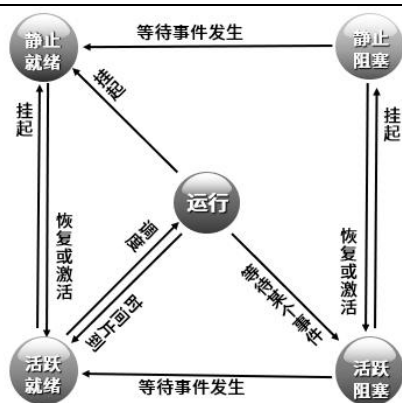
【要点分析】

操作系统三态模型如下图所示：



注：运行→等待，等待→就绪，前者等待某个事件，后者是等待的这个事件发生了，注意区分。

操作系统五态模型：



【备考点拨】

1、掌握操作系统三态模型、五态模型的状态位置及其状态变迁条件。

2.2 前趋图 (★★★★)

【考法分析】

本考点主要的考查形式有：与 PV 操作结合考查。

【要点分析】

1、前趋活动和后继活动：在前趋图中，前趋活动完成后通知所有后继活动；后继活动开始之前要检查是否前趋活动已经全部完成。

【备考点拨】

1、掌握前趋图相关表示的活动之间的依赖关系。

2.3 信号量与 PV 操作 (★★★★)

【考法分析】

本知识点的考查形式有：单独考查信号量与 PV 操作控制进程的互斥和并发；结合前趋图考查某个位置对应的 P、V 操作及其信号量；根据题干描述的业务逻辑判断对应位置的 P、V 操作及其信号量。

【要点分析】

1、相关概念：互斥、同步、临界资源、临界区、信号量。

互斥：如千军万马过独木桥，同类资源的竞争关系。

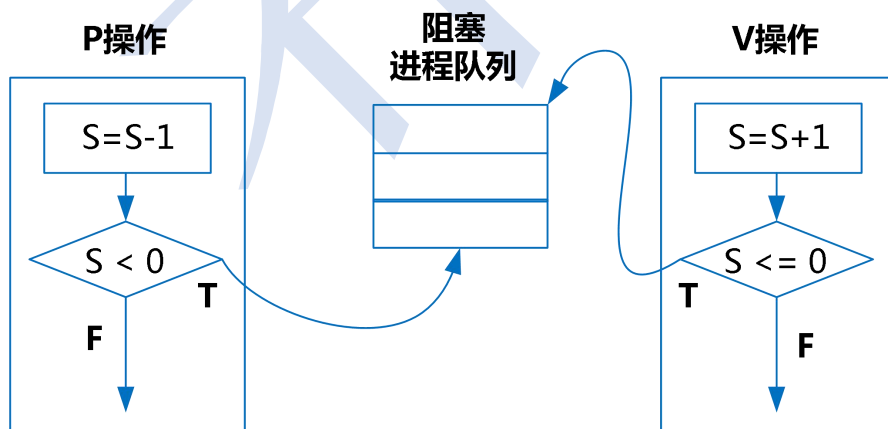
同步：速度有差异，在一定情况停下等待，进程间的协作关系。

临界资源：诸进程间需要互斥方式对其进行共享的资源，如打印机、磁带机等

临界区：每个进程中访问临界资源的那段代码称为临界区

信号量：是一种特殊的变量，表示资源数。当信号量小于 0 时，还可以表示排队进程数。

2、PV 操作对应的过程（如下图所示）：



3、前趋图与 PV 操作结合，根据前趋图箭线标注信号量，再根据进程图填空：

针对箭线标注信号量，箭线的起点位置是 V 操作（即前趋活动结束后以 V 操作通知后继活动）；箭线的终点位置是 P 操作（即后继活动开始前以 P 操作检查前趋活动是否完成）。

【备考点拨】

- 1、掌握 PV 操作及信号量相关的概念；
- 2、理解 PV 操作的原理和应用，学会利用相关解题技巧，解决 PV 操作与信号量、前趋图的结合考查。

2.4 死锁及银行家算法 (★★★★)

【考法分析】

本知识点主要考查形式有：根据进程情况计算死锁资源数；根据资源数利用银行家算法进行资源分配，判断选项中给出的序列是否安全。

【要点分析】

1、了解死锁的条件和预防概念：进程管理是操作系统的核心，但如果设计不当，就会出现死锁的问题。如果一个进程在等待一件不可能发生的事，则进程就死锁了。而如果一个或多个进程产生死锁，就会造成系统死锁。

2、根据题干给出的进程和资源分配，判断形成死锁的最小资源数或其他参数：对于这种情况，分配资源时每个进程得到可以完成进程的资源数减一，此时是形成死锁的最差情况，在此情况下多 1 个资源即可解决死锁问题，即不可能形成死锁。假设 m 个进程各自需要 w 个 R 资源，系统中共有 n 个 R 资源，此时不可能形成死锁的条件是：

$$m \times (w-1) + 1 \leq n.$$

3、银行家算法：当一个进程对资源的最大需求量不超过系统中的资源数时可以接纳该进程。进程可以分期请求资源，但请求的总数不能超过最大需求量。当系统现有的资源不能满足进程尚需资源数时，对进程的请求可以推迟分配，但总能使进程在有限的时间里得到资源。根据银行家算法判断相关进程序列是否会形成死锁，是则为不安全序列。

【备考点拨】

- 1、掌握相关概念；
- 2、掌握死锁资源数计算；
- 3、掌握银行家算法分配资源时判断安全序列。

【存储管理】

2.5 段页式存储 (★★★★)

【考法分析】

本知识点主要考查形式有：页式存储中对应逻辑页的物理页号，或对应逻辑地址的物理地址；段式存储中对应段地址的合法性判断；页式存储、段式存储、段页式存储的一些概念描述判断正误。

【要点分析】

- 1、知道页面大小时，可以依此判断页内地址的长度，并据此知道该地址的页号；
- 2、页号与页帧号的转换可以通过查表进行；
- 3、段地址的格式，段号后跟段内地址不能超过段长；
- 4、页式存储：将程序与内存均划分为同样大小的块，以页为单位将程序调入内存。
- 5、段式存储：按用户作业中的自然段来划分逻辑空间，然后调入内存，段的长度可以不一样。
- 6、段页式存储：段式与页式的综合体。先分段，再分页。1 个程序有若干个段，每个段中可以有若干页，每个页的大小相同，但每个段的大小不同。

【备考点拨】

- 1、掌握段页式存储相关的一些概念；
- 2、掌握页式存储地址的转换和页表的查找；
- 3、掌握段式存储段地址合法性判断。

2.6 页面置换算法 (★)

【考法分析】

本知识点主要与页式存储结合考查，依据最近最少被使用原则选择应该被淘汰的页面。

【要点分析】

- 1、页面淘汰时，主要依据原则（考试中默认按照此原则进行淘汰）：先淘汰最近未被访问的（访问位为 0），其次淘汰但未被修改的（即修改位为 0，因为修改后的页面淘汰时代价更大）。
- 2、页面淘汰算法有多种，常用的是 LRU 即最近最少使用原则，依据的是局部性原理。
- 3、对于多种淘汰算法：最优算法 OPT（理想型），随机算法 RAND（随机性），先进先出 FIFO（可能产生“抖动”），最近最少使用 LRU（依据局部性原理）。

【备考点拨】

- 1、掌握页表字段表示的意义，根据 LRU 进行页面淘汰；
- 2、了解多种淘汰算法的原则，根据它们的特点进行区分。

2.7 磁盘管理 (★★)

【考法分析】

本知识点的考查形式有：计算磁盘数据的读取时间；优化存储后的数据读取时间；磁盘调度算法的相关应用。

【要点分析】

1、存取时间=寻道时间+等待时间，寻道时间是指磁头移动到磁道所需的时间；等待时间为等待读写的扇区转到磁头下方所用的时间。有时还需要加上数据的传输时间。

2、在处理过程中，如果有关于缓冲区的使用，需要了解对于单缓冲区每次只能被一个进程使用，即向缓冲区传输数据的时候不能从缓冲区读取数据，反之亦然。

3、对于磁盘存储的优化，是因为磁头保持转动的状态，当读取数据传输或处理时，磁头会移动到超前的位置，需要继续旋转才能回到逻辑下一磁盘块，优化存储就是调整磁盘块的位置，让逻辑下一磁盘块放到磁头将要开始读取该逻辑块的位置。

4、磁盘调度算法（考虑移臂时，只需要考虑柱面信息即可，扇区不需要考虑，同一柱面的多个扇区先后顺序随机）：

先来先服务 FCFS（谁先申请先服务谁）；

最短寻道时间优先 SSTF（申请时判断与磁头当前位置的距离，谁短先服务谁）；

扫描算法 SCAN（电梯算法，双向扫描）；

循环扫描 CSCAN（单向扫描）。

【备考点拨】

- 1、掌握读取磁盘数据时间计算方法；
- 2、掌握磁盘存储优化的过程和计算方法；
- 3、了解磁盘调度算法的区别，并能加以区分。

【文件管理】

2.8 绝对路径与相对路径 (★★★★)

【考法分析】

本知识点主要考查形式即给出图示，要求选择正确的绝对路径、相对路径、文件全名。

【要点分析】

- 1、绝对路径从根目录开始写起，并且该文件的全名即为绝对路径+文件名。
- 2、相对路径从当前位置下一级目录开始写起。

【备考点拨】

- 1、掌握绝对路径和相对路径的写法。
- 2、掌握树形目录的一些特点和概念。

2.9 索引文件 (★★)

【考法分析】

本知识点的考查形式主要是具体逻辑块号的索引方式判断，以及索引方式所能表示的文件大小，中间会涉及到计算。

【要点分析】

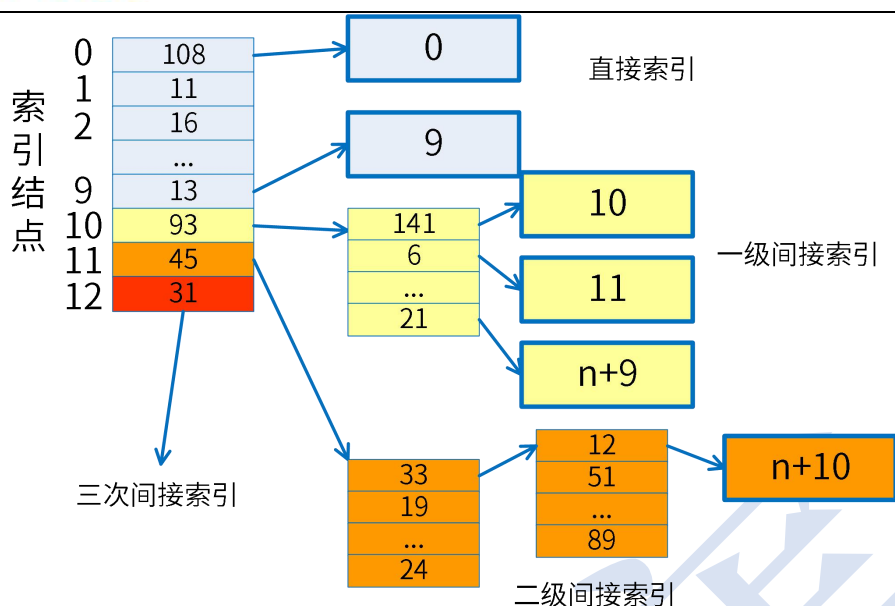
1、索引结点对应的索引方式一般题干会给出，没有给出的默认按照如图所示方式理解，下面的文件大小依图给出计算过程。

2、根据物理块大小（假设 1KB）和地址项长度（假设 4B），可以计算存放间接索引的物理块可以存放的地址项个数：物理块大小/地址项长度，向下取整（1KB/4B=256，注意单位和进制转换）。

3、直接索引（即索引结点直接指向实际存储文件的物理块），能够表示的逻辑页号范围是 0~9，能够表示的文件大小时 10*1KB。

4、一级间接索引（即索引结点指向的物理块存放的是地址项，对应地址项个数 256 个，可以指向 256 个实际存储文件的物理块），能够表示的逻辑页号范围是 10~265，能够表示的文件大小是 256*1KB。

5、二级间接索引（即索引结点指向的物理块存放的是间接索引的地址项，共 256 个，可以指向 256 个存放地址项的物理块，每个物理块指向实际存储文件的地址项有 256 个，最终指向的物理块共有 256*256 个），能够表示的逻辑页号范围是 266~65801，能够表示的文件大小是 65536KB。



【备考点拨】

- 1、掌握索引文件的具体对应关系及相关的一些概念描述；
- 2、掌握索引文件逻辑页号和物理块的对应关系；
- 3、掌握索引文件表示文件长度的计算。

2.10 位示图 (★★)

【考法分析】

本知识点的主要考查方式是计算指定磁盘存放的对应字的序号或位置。

【要点分析】

- 1、对于位示图，每一个 bit 位可以表示一个磁盘的占用情况，“0”表示空闲，“1”表示占用。
- 2、对于字的长度与具体机器字长有关，有题目指定，假设机器字长 16 位，则每个字可以表示 16 个磁盘块的占用情况；
- 3、指定序号为 n 或第 $n+1$ 个磁盘，占用情况需要用 $m = (n+1)/16$ （向上取整）个字表示，字的序号为 $m-1$ 。注意其中磁盘序号、字的序号、对应位号都是从 0 开始，计算过程中会有加 1 或减 1 处理。

【备考点拨】

- 1、掌握相关的概念；
- 2、掌握相关的计算过程。

【设备管理】

2.11 I/O 设备管理 (★)

【考法分析】

本知识的考查形式主要是给出一些描述判断对应的传输方式，或关于 I/O 系统的层次对应位置。

【要点分析】

- 1、对于 I/O 传输控制方式：

程序查询方式（CPU 一直处于询问、等待的过程，占用 CPU 时间最长，CPU 利用率最低）；

中断方式（I/O 完成后向 CPU 发送中断请求信号，CPU 和 I/O 可以并行）；

DMA（CPU 只做初始化，不参与具体数据传输过程）；

通道方式、I/O 处理机，专用硬件方式。

- 2、对于 I/O 软件：



硬件：完成具体的 I/O 操作。

中断处理程序：I/O 完成后唤醒设备驱动程序

设备驱动程序：设置寄存器，检查设备状态

设备无关 I/O 层：设备名解析、阻塞进程、分配缓冲区

用户级 I/O 层：发出 I/O 调用。

【备考点拨】

1、掌握 I/O 传输控制方式的特点，能够加以区分；

2、了解 I/O 软件的层次和相对位置。

3 章节问答

1、pv 操作都必须要成对出现吗？

答：

PV 一定是成对出现的，PV 可以简单理解为加锁和解锁，如果只加锁也就是只有 P 操作，可能会形成死锁。如果没有加锁，只有解锁也就是 V 操作，那么相当于系统中资源无端增多，也就没有意义了。

以打印机为例，如果对打印机加上 PV 操作，P 操作相当于，每个人用之前都锁定一次打印机 ($S=S-1$)，此时其他人去打印时进行申请，会发现资源不足 ($S<0$)，此时就进入阻塞队列去排队了；而 V 操作相当于，能够使用打印机的人，用完之后，之前占有的打印机就会释放给其他人用 ($S=S+1$)，此时如果发现有人排队 ($S<=0$)，就会通知排队的人，可以使用了。

如果只占有不释放，很明显，排队的人只会持续暴增；如果只释放，那么信号量 S 无端增加 1，而不会减少，也就没有意义了。

2、PV 操作与互斥、同步模型有什么联系？

答：

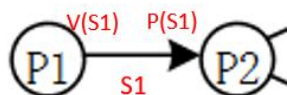
在互斥模型中，为了控制对临界资源的访问，信号量初值是临界资源的资源数，每一次使用临界资源前都会用 P 操作对资源进行申请锁定，用完之后立马 V 操作对资源进行释放，如果有进程排队，此时还会对阻塞队列进行唤醒操作。一般来说，互斥模型的 PV 操作，信号量针对的是临界资源，称为互斥信号量，初值为临界资源初始数量，在同一个进程中有成对的 PV 控制访问。

在同步模型中，为了协调多个进程，PV 操作会分布在不同的进程中，此时的信号量称为同步信号量，同步模型会有等待的过程，此时的信号量初值经常会出现 0 的情况，也就是初始资源不足，等待其他进程制造资源并唤醒。

3、PV 操作与前趋图结合考查时，为什么存在信号量比进程数还多的情况？

答：

PV 操作与前趋图结合考查时，前趋图一般是同步模型，为了协调控制，此时信号量针对的不是进程，而是进程的变迁，也就是说，信号量的数量与箭线的数量是相等的，PV 操作控制的是进程变迁过程。



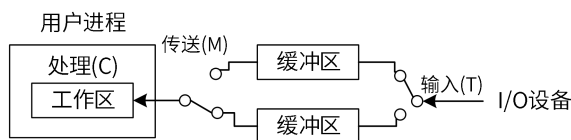
4、移臂调度过程中，同一柱面有多个扇区怎么办？

答：

一般而言，涉及到移臂调度算法时，只需要考虑移臂过程，查看柱面顺序即可。旋转等待时间跟多磁头（盘面）是否并发读写有关系，从目前磁盘技术来看，是支持多个盘面同时读写的。所以只要是同一柱面，扇区顺序没有要求。

5、磁盘读取数据过程中单缓冲区和双缓冲区有什么区别？

答：磁盘读取涉及到缓冲区一般形式如下：



处理数据可以理解三个处理步骤：①从磁盘读入到缓冲区；②从缓冲区读入到（内存）用户区；③处理（内存）用户区数据。

如果是单缓冲区，则意味步骤①和②都需要访问临界资源——缓冲区，所以需要合并成一个操作阶段。（缓冲区同一时刻只允许一个进程访问）

如果是双缓冲区，可以实现读入到缓冲区 2 和从缓冲区 1 读入到用户区的并发。

可以构造流水线进行计算。

单缓冲区：使用缓冲区时不能并行，必须分开执行，时间为①和②的时间之和；然后处理数据。构造成流水线后，整个过程划分为 2 个阶段，分别是①和②的时间之和，③的时间，根据流水线执行公式进行计算。

双缓冲区：读入缓冲区，和从缓冲区读入用户区，可以对不同的缓冲区进行，也就是说，可以并行处理。对于这里构造成流水线后，整个过程划分为 3 个阶段，即①、②、③。根据流水线执行公式进行计算结课。

PS：注意理解这里流水线的构造过程。

第 3 章 数据库系统

1 考情分析

根据对历年的考试真题进行分析，本章要求考生掌握以下几个方面的知识：

- (1) 了解数据库模式、数据库设计过程。
- (2) 熟悉 E-R 图图示，掌握 E-R 图转关系模式的原则。
- (3) 掌握关系代数（并、交、差、笛卡尔积、投影、选择、自然连接）。
- (4) 熟悉函数依赖、部分函数依赖、传递函数依赖等概念，熟悉候选码\候选键、主码\主键、外键的概念，掌握候选码的求取，区分主属性和非主属性概念。
- (5) 了解规范化相关的问题描述和作用，掌握 1NF、2NF、3NF、BCNF 的判断区分。
- (6) 掌握模式分解对于无损、保持函数依赖的判断。
- (7) 了解常用的 SQL 语言语法，比如 select 用法。
- (8) 了解事务的特性，事务并发可能产生的问题，封锁协议（S 锁和 X 锁）。
- (9) 了解数据完整性分类（数据完整性、参照完整性、自定义完整性等）。
- (10) 了解分布式数据库、数据库备份相关扩展知识。
- (11) 掌握软件设计-数据库设计部分（下午题试题二）解题技巧。

1.1 本章重点

序号	知识领域	知识点详情
1	数据库模式 (★★)	数据库模式
1	数据库设计过程 (★★)	数据库设计过程
1	ER 模型 (★★★★★)	E-R 图图元
2		E-R 图转关系模式原则
1	关系代数 (★★★★)	关系代数 (并交叉、笛卡尔积、投影、选择、自然连接)
1	规范化理论 (★★★★★)	规范化理论

1	SQL 语言 (★★★★)	SQL 语言
1	并发控制 (★★)	并发控制
1	数据库完整性约束 (★)	数据库完整性约束 (★)
1	数据库扩展知识 (★)	分布式数据库
2		数据库备份
1	【软件设计】 数据库设计题解题技巧	数据库设计题解题技巧

2 考点精讲

2.1 数据库模式 (★★)

【考法分析】

本知识点的主要考查方式是判断模式（外模式、模式、内模式）与产物（视图、库表、文件）的对应关系，或给定一些概念描述判断正误。

【要点分析】

1、三级模式：外模式对应视图，模式（也称为概念模式）对应数据库表，内模式对应物理文件。

2、两层映像：外模式-模式映像，模式-内模式映像；两层映像可以保证数据库中的数据具有较高的逻辑独立性和物理独立性。

3、逻辑独立性：即逻辑结构发生改变时，用户程序对外模式的调用可以不做修改；物理独立性：即数据库的内模式发生改变时，数据的逻辑结构不变。

【备考点拨】

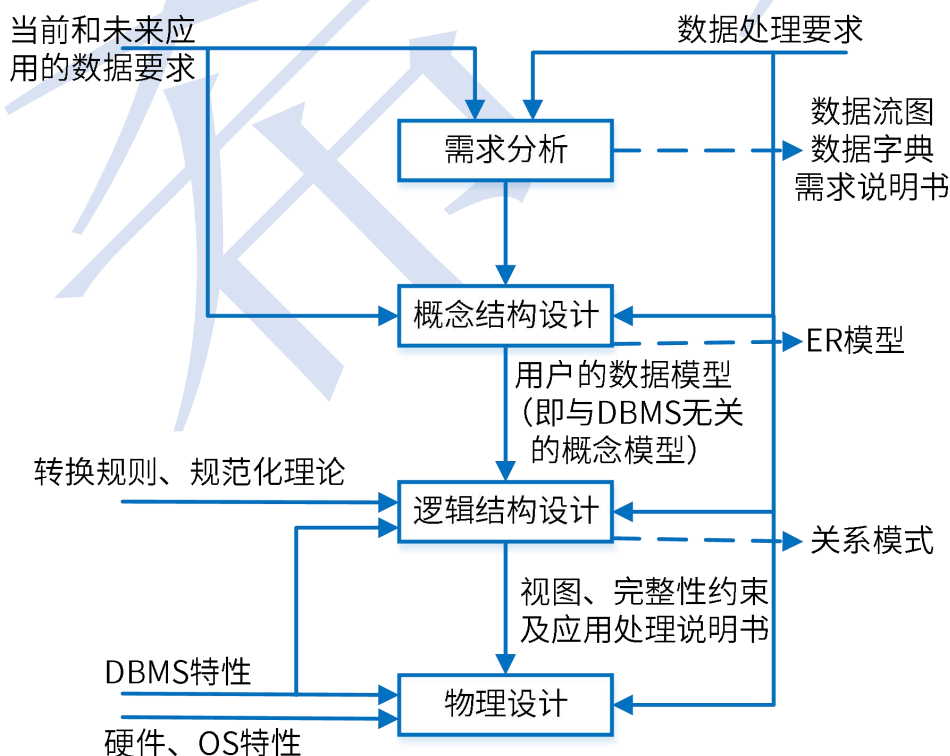
掌握三级模式-两层映像结构和相关概念。

2.2 数据库设计过程 (★★)

【考法分析】

本知识点主要考查形式有：在上午题中判断各个阶段的任务和产物；在下午题中数据库设计题考查思路与数据库设计过程基本一致。

【要点分析】



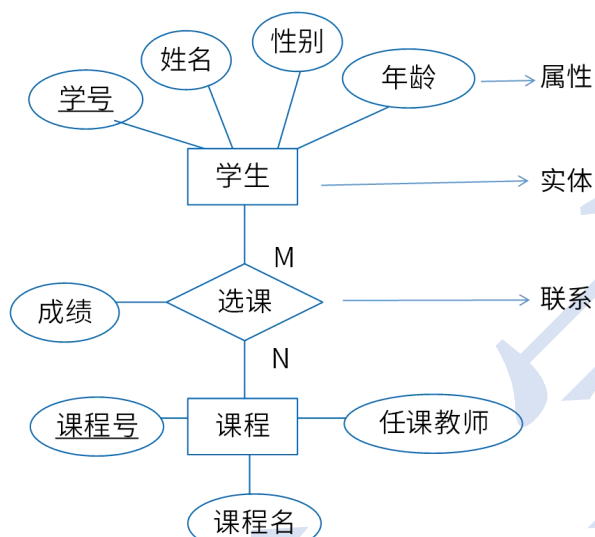
2.3 ER 模型 (★★★★★)

【考法分析】

本知识点主要考查形式有：在上午题中给出 E-R 图让考生判断某些部分的缺失、定义，或关系的类型判断，E-R 图向关系模式的转换；在下午题中数据库设计题会考查补充 E-R 图，并且会涉及到 E-R 图向关系模式转换。

【要点分析】

2.3.1 E-R 图图元



1、**实体**：在 E-R 模型中，实体用矩形表示，通常矩形框内写明实体名。实体是现实世界中可以区别于其他对象的“事件”或“物体”。实体集是具有相同属性的实体集合。我们常见的实体意义应该是实体集。（比如某一个学生是实体，所有学生都具有一些相同的属性，因此学生的集合可以定义为一个实体集。也可以对应到具体数据库，实体集是表，而实体是其中特定的条目元组）

2、**属性**：属性是实体某方面的特性。在 E-R 图中用椭圆表示，并以无向边与对应实体进行连接。每个属性都有其取值范围。属性具有以下分类：

简单属性和复合属性。单独属性是原子的、不可再分的，复合属性可以分解为更小的部分。一般对于复合属性会在题干中给出其分解特性，比如对于地址，可进一步划分为省、市、区，若不特别声明，通常指的是简单属性。

单值属性和多值属性。对于一个特定的实体都只有一个特定的值，则为单值属性；在某些特殊的情况下，一些属性可能对应一组值，这样的属性为多值属性。比如对于职工家属，可能有多名成员，会有一组取值，这里即为多值属性。

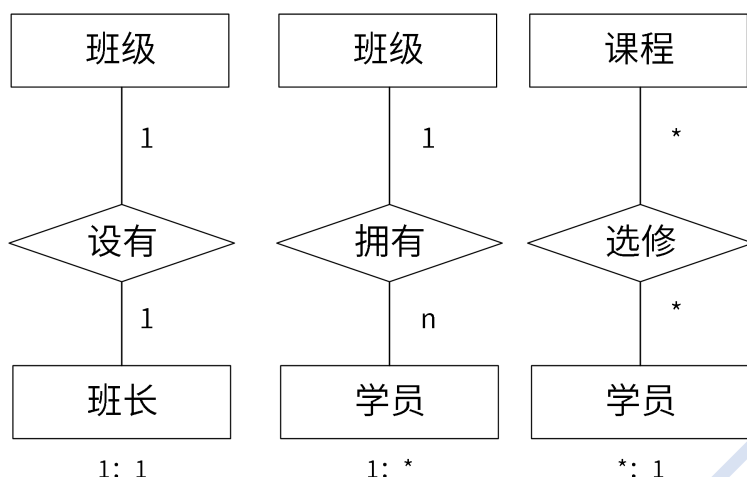
派生属性。派生属性可以从其他属性得来。比如出生年月可以计算得出年龄，当出生年月属性已记录时，年龄可以计算得来，为派生属性。

NULL 属性。当实体在某个属性上没有特定值或属性值未知时，使用 NULL 值，表示无异议或不知道。

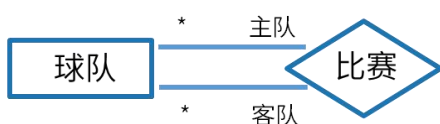
3、联系

在 E-R 模型中，联系用菱形表示，通常菱形框内写明联系名，并用无向边分别与有关实体连接起来，同时在无向边旁标注上联系的类型（1:1、1:n、n:m）。

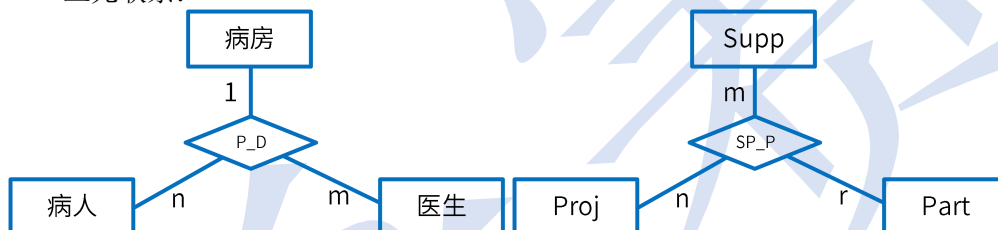
不同实体集间的二元关系：



同一实体集内部的二元联系：



三元联系：



以三元关系中的一个实体作为中心，假设另两个实体都只有一个实例：

若中心实体只有一个实例能与另两个实体的一个实例进行关联，则中心实体的连通数为“一”。

若中心实体有多于一个实例能与另两个实体实例进行关联，则中心实体的连通数为“多”。

语义表示：

P_D：表示一个病房有多个病人和多个医生，一个医生只负责一个病房，一个病人只属于一个病房。

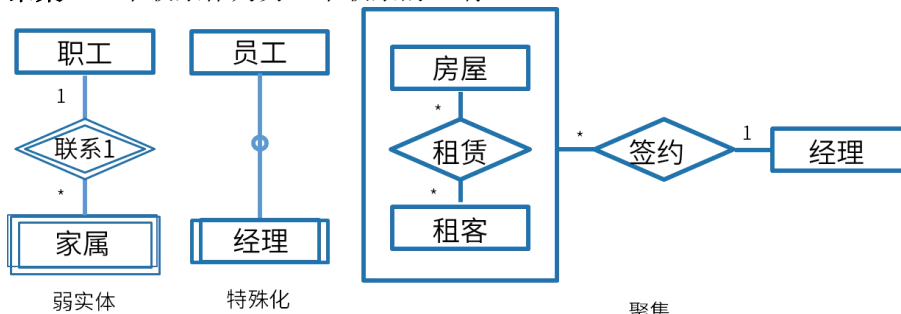
SP_P：表示供应商为多个项目供应多种零件，每个项目可用多个供应商供应的零件，每种零件可由不同的供应商供应。

4、扩展的 E-R 模型

弱实体：在现实世界中有一种联系比较特殊，这种联系代表实体间的所有（Ownership）关系。这种实体对于另一些实体存在很强的依赖关系，即一个实体的存在必须以另一个实体为前提，将这类实体称为弱实体。比如附件是邮件的弱实体。弱实体一般以双矩形表示，与其依赖的实体之间存在联系：

特殊化：特殊化主要描述的是子类 and 父类之间的关系。子类是超类的特殊化，子类继承超类的所有属性和联系，并且还有自己特殊的属性和联系。比如经理是员工的特殊化。特殊化实体一般以双边矩形表示，与其超类以线圈的方式连接：

聚集：一个联系作为另一个联系的一端。



2.3.2 E-R 图转关系模式原则

1、每个实体型转换为一个关系模式

2、联系转关系模式：

(1) 一对一联系的转换有两种方式。

独立的关系模式： 并入两端主键及联系自身属性。(主键：任一端主键)

归并(任意一端)： 并入另一端主键及联系自身属性。(主键：保持不变)

(2) 一对多联系的转换有两种方式。

独立的关系模式： 并入两端主键及联系自身属性。(主键：多端主键)

归并(多端)： 并入另一端主键及联系自身属性。(主键：保持不变)

(3) 多对多联系的转换只有一种方式

独立的关系模式： 并入两端主键及联系自身属性。(主键：两端主键的组合键)

【备考点拨】

1、掌握 E-R 图的绘制；

2、能够正确识别实体、弱实体、属性、联系、联系类型；

3、掌握 E-R 图向关系模式的转换。

2.4 关系代数(★★★)

【考法分析】

本知识点主要考查形式是：给定代数式，求取计算结果或其结果的特性，找到等价表达式，常考的关系代数是笛卡尔积、选择、投影组合与自然连接的等价表示，以及与 SQL 语句 SELECT 结合考查。

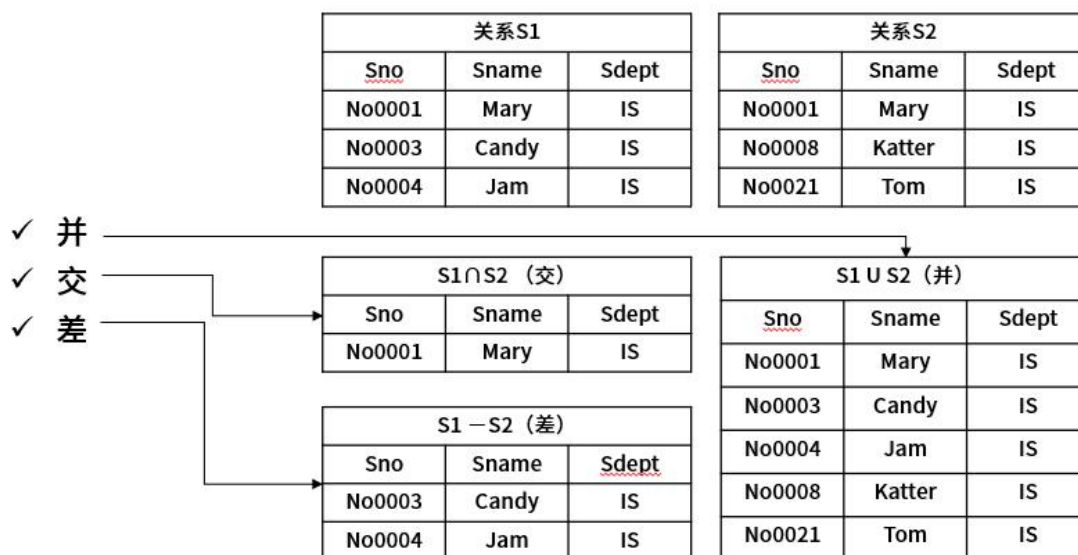
【要点分析】

1、并(结果为二者元组之和去除重复行)

2、交(结果为二者重复行)

3、差(前者去除二者重复行)

类似于集合运算，计算如下图所示：



4、笛卡尔积：结果列数为二者属性列数之和，行数为二者元素数乘积。

5、投影：对属性列的选择列出。

6、选择：对元组行的选择列出。

属性名可以依次标序号，直接以数字形式出现在表达式中。计算如下图所示：

S1 × S2 (笛卡尔积)					
Sno	Sname	Sdept	Sno	Sname	Sdept
No0001	Mary	IS	No0001	Mary	IS
No0001	Mary	IS	No0008	Katter	IS
No0001	Mary	IS	No0021	Tom	IS
No0003	Candy	IS	No0001	Mary	IS
No0003	Candy	IS	No0008	Katter	IS
No0003	Candy	IS	No0021	Tom	IS
No0004	Jam	IS	No0001	Mary	IS
No0004	Jam	IS	No0008	Katter	IS
No0004	Jam	IS	No0021	Tom	IS

关系S1		
Sno	Sname	Sdept
No0001	Mary	IS
No0003	Candy	IS
No0004	Jam	IS

关系S2		
Sno	Sname	Sdept
No0001	Mary	IS
No0008	Katter	IS
No0021	Tom	IS

$\pi_{Sno, Sname}(S1)$ (投影)	
Sno	Sname
No0001	Mary
No0003	Candy
No0004	Jam

$\sigma_{Sno=No0003}(S1)$ (选择)		
Sno	Sname	Sdept
No0003	Candy	IS

7、自然连接：结果列数为二者属性列数之和减去重复列，行数为二者同名属性列其值相同的结果元组。
笛卡尔积、选择、投影的组合表示可以与自然连接等价。
普通连接的条件会写出，没有写出则表示为自然连接。计算如图所示：

关系S1		
Sno	Sname	Sdept
No0001	Mary	IS
No0003	Candy	IS
No0004	Jam	IS

关系S2	
Sno	Age
No0001	23
No0008	21
No0021	22

S1 ⋈ S2			
Sno	Sname	Sdept	Age
No0001	Mary	IS	23

$$\pi_{1, 2, 3, 5}(\sigma_{1=4}(S1 \times S2))$$

【备考点拨】

- 1、掌握关系代数的计算。
- 2、关系代数经常与 SELECT 语句结合考查。对于常规 SELECT 语句格式：SELECT 语句 1 FROM 语句 2 WHERE 语句 3，其中语句 1 是投影列的列名，语句 2 是查询的表格对象（如果有多个表格，用逗号隔开，则结果从笛卡尔积处理），语句 3 为选择行的条件。

2.5 规范化理论 (★★★★★)

【考法分析】

本知识点主要的考查形式是找出某些关系中的主键、外键，判断某些关系模式的规范化程度，有时会在下午题中出现相关的提问，涉及到概念的考查。

【要点分析】

1、**候选键（候选码）**是能够唯一标示元组却无冗余的属性组合，可以有多种不同的候选键，在其中任选一个作为**主键**。候选键的求取可以利用图示法找入度为 0 的属性集合，并在此基础上进行扩展，最终找到能够遍历全图的最小属性组合作为候选键，对于入度为 0 在关系依赖集中可以理解为从未在箭头右侧出现。

2、组成候选码的属性就是**主属性**，其他为**非主属性**。

3、**外键**是其他关系模式的主键。

4、**范式**：规范化过程是为了解决数据冗余、删除异常、插入异常、更新异常等问题。

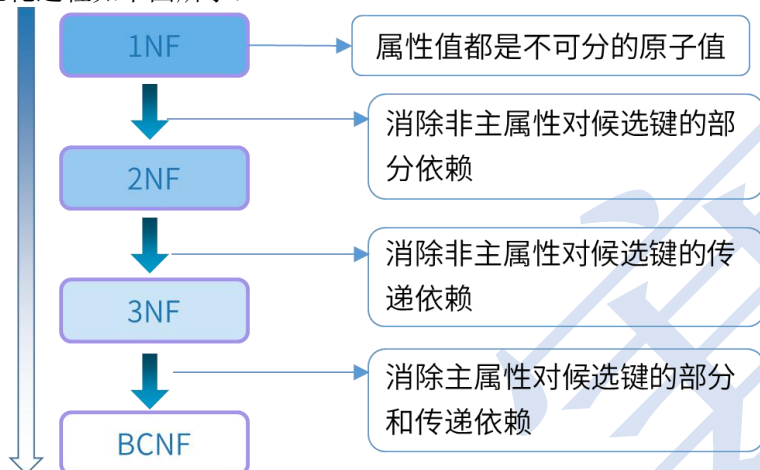
(1) **第一范式 (1NF)**：在关系模式 R 中，当且仅当所有域只包含原子值，即每个属性都是不可再分的数据项，则称关系模式 R 是第一范式。

(2) **第二范式 (2NF)**：当且仅当关系模式 R 是第一范式 (1NF)，且每一个非主属性完全依赖候选键（没有不完全依赖）时，则称关系模式 R 是第二范式。

(3) **第三范式 (3NF)**：当且仅当关系模式 R 是第二范式 (2NF)，且 R 中没有非主属性传递依赖于候选键时，则称关系模式 R 是第三范式。

(4) **BC 范式 (BCNF)**：设 R 是一个关系模式，F 是它的依赖集，R 属于 BCNF 当且仅当其 F 中每个依赖的决定因素必定包含 R 的某个候选码。

规范化过程如下图所示：



5、**规范化过程**：分解关系模式。

(1) **保持函数依赖**：设数据库模式 $\rho = \{R_1, R_2, \dots, R_k\}$ 是关系模式 R 的一个分解，F 是 R 上的函数依赖集， ρ 中每个模式 R_i 上的 FD 集是 F_i 。如果 $\{F_1, F_2, \dots, F_k\}$ 与 F 是等价的（即相互逻辑蕴涵），那么称分解 ρ 保持 FD。

(2) **无损联接分解**：指将一个关系模式分解成若干个关系模式后，通过自然联接和投影等运算仍能还原到原来的关系模式。（表格法，公式法—仅限分解为 2 个子关系）

【备考点拨】

- 1、掌握候选键、主键、外键的求取；
- 2、掌握规范化理论相关的概念和规范化过程；
- 3、掌握关系分解后，对无损分解、保持函数依赖的判断。

2.6 SQL 语言 (★★★★)

【考法分析】

本知识点的考查形式主要有：与关系代数结合考查相关 SQL 语言的写法，或单纯考查 SQL 语言的应用。

【要点分析】

1、SQL 语言

分类	动词
数据查询	SELECT
数据定义	CREATE、DROP、ALTER
数据操纵	INSERT、UPDATE、DELETE

数据控制	GRANT、REVOKE
------	--------------

2、简单查询语句

SELECT [ALL|DISTINCT] <目标表达式> [, <目标表达式>]...

FROM <表名> [, <表名>]...

[WHERE <条件表达式>]

[ORDER BY <列名 2> [ASC|DESC] ...];

3、分组查询

[GROUP BY <列名 1> [HAVING <条件表达式>]]

聚集函数

处理类型	处理子类	示例/语法
结果排序	升序或降序	ORDER BY 字段名 DESC ASC
集函数	统计	COUNT([DISTINCT ALL] <列名>)
	计算一列中值的总和	SUM([DISTINCT ALL] <列名>)
	计算一列值的平均值	AVG([DISTINCT ALL] <列名>)
	求一列值中的最大值	MAX([DISTINCT ALL] <列名>)
	求一列值中的最小值	MIN([DISTINCT ALL] <列名>)
对结果分组	将查询结果按列值分组	GROUP BY <列名>
对分组结果筛选	对分组结果筛选	HAVING <条件列达式>

4、权限控制语句

授权语句 GRANT <权限>[, ... n]

ON <对象类型><对象名>

TO <用户>[, ... n]

WITH GRANT OPTION

WITH GRANT OPTION 子句，获得权限的用户还可以将权限赋给其他用户

收回权限语句 REVOKE<权限>[, ... n]

ON <对象类型><对象名>

FROM <用户>[, ... n]

[RESTRICT|CASCADE]

【备考点拨】

1、掌握常用的 SQL 语言的语法。

2.7 并发控制 (★★)

【考法分析】

本知识点的考查形式主要是给出一些情景判断出现的并发问题，或给出一些关于事务、锁等概念的描述，判断正误。

【要点分析】

1、事务特性（ACID）：

原子性：事务是原子的，要么都做，要么都不做。

一致性：事务执行的结果必须保证数据库从一个一致性状态变到另一个一致性的状态。因此，当数据库只包含成功事务提交的结果时，称数据库处于一致性状态。

隔离性：事务相互隔离，当多个事务并发执行时，任一事务的更新操作直到其成功提交的整个过程，对其他事务都是不可见的。

持续性：一旦事务成功提交，即使数据库崩溃，其对数据库的更新操作也将永久有效。

2、并发产生的问题：

(1) 丢失更新/丢失修改		(2) 不可重复读		(3) 读“脏”数据	
T1	T2	T1	T2	T1	T2
①读 A=10 ② ③A=A-5 写回 ④	读 A=10 A=A-8 写回	①读 A=20 读 B=30 求和=50 ② ③读 A=70 读 B=30 求和=100 (验算不对)	读 A=20 A←A+50 写 A=70	①读 A=20 A←A+50 写回 70 ② ③ROLLBACK A 恢复为 20	读 A=70

3、封锁协议：

共享锁（S锁）：若事务 T 对数据对象 A 添加了 S 锁，则只允许 T 读取 A，但不能修改 A。并且其他事务只能对 A 加 S 锁，不能加 X 锁。

排他锁（X锁）：若事务 T 对数据对象 A 添加了 X 锁，则只允许 T 读取和修改 A，其他事务不能再对 A 加任何锁。

【备考点拨】

- 1、掌握事务特性的概念。
- 2、能够区分并发产生的问题。
- 3、了解封锁协议的加锁原则和两种锁的特性。

2.8 数据库完整性约束（★）

【考法分析】

本知识点的考查形式主要是给出一定描述，判断其正误。

【要点分析】

实体完整性：规定其主属性不能去空值。

参照完整性（也称为引用完整性）：规定其外键为参照表的主键值或为空值。

用户自定义完整性：指用户针对某一具体的关系数据库的约束条件，反映某一具体应用所涉及的数据必须满足的予以要求，由应用的环境决定，如年龄定义为 0~150 正整数。

触发器：一种复杂的完整性约束。

【备考点拨】

- 1、掌握完整性约束相关的概念，能够区分不同的完整性类别，判断一些描述的正误。

2.9 数据库扩展知识（★）

【考法分析】

本知识点的考查形式主要是给出一定描述，判断其对应概念。

【要点分析】

- 1、**数据库备份分类：**从备份量来分，可以分为完全备份、增量备份、差异备份。

完全备份：备份所有数据。即使两个备份时间点之间数据没有任何变动，所有数据还是会被备份下来。

增量备份：跟完全备份不同，增量备份在做数据备份前会先判断数据的最后修改时间是否比上次备份的时间晚。如果不是，则表示该数据并没有被修改过，这次不需要备份。所以该备份方式，只记录上次备份之后的变动情况，而非完全备份。

差异备份：差异备份与增量备份一样，都只备份变动过的数据。但前者的备份是针对上次完整备份后，曾被更新过的。

2、分布式数据库透明性

分片透明：是指用户不必关心数据是如何分片的，它们对数据的操作在全局关系上进行，即如何分片对用户是透明的，因此，当分片改变时应用程序可以不变。分片透明性是最高层次的透明性，如果用户能在全局关系一级操作，则数据如何分布，如何存储等细节自不必关心，其应用程序的编写与集中式数据库相同。

复制透明：用户不用关心数据库在网络中各个节点的复制情况，被复制的数据的更新都由系统自动完成。在分布式数据库系统中，可以把一个场地的数据复制到其他场地存放，应用程序可以使用复制到本地的数据在本地完成分布式操作，避免通过网络传输数据，提高了系统的运行和查询效率。但是对于复制数据的更新操作，就要涉及到对所有复制数据的更新。

位置透明：是指用户不必知道所操作的数据放在何处，即数据分配到哪个或哪些站点存储对用户是透明的。

局部映像透明性（逻辑透明）：是最低层次的透明性，该透明性提供数据到局部数据库的映像，即用户不必关心局部 DBMS 支持哪种数据模型、使用哪种数据操纵语言，数据模型和操纵语言的转换是由系统完成的。因此，局部映像透明性对异构型和同构异质的分布式数据库系统是非常重要的。

3、分布式数据库特点

数据独立性。除了数据的逻辑独立性与物理独立性外，还有数据分布独立性（分布透明性）。

集中与自治共享结合的控制结构。各局部的 DBMS 可以独立地管理局部数据库，具有自治的功能。同时，系统又设有集中控制机制，协调各局部 DBMS 的工作，执行全局应用。

适当增加数据冗余度。在不同的场地存储同一数据的多个副本，可以提高系统的可靠性和可用性，同时也能提高系统性能。

（提高系统的**可用性**，即当系统中某个节点发生故障时，因为数据有其他副本在非故障场地上，对其他所有场地来说，数据仍然是可用的，从而保证数据的完备性。）

全局的一致性、可串行性和可恢复性。

【备考点拨】

- 1、了解备份的概念以及不同备份方式的区别；
- 2、了解分布式数据库的特点和透明性的描述。

2.10 【软件设计】数据库设计题解题技巧

1、补充 E-R 图

找实体：实体一般为名词形式，实体是现实世界中可以区别于其他对象的“事件”或“物体”。在 E-R 图中可根据相关实体间的联系，补充缺失的实体。实体以矩形表示，在矩形框内写明实体名。

找联系：在 E-R 模型中，联系用菱形表示，通常菱形框内写明联系名，并用无向边分别与有关实体连接起来，同时在无向边旁标注上联系的类型（1:1、1:n、n:m）。根据题干描述找出对应实体间存在的联系。

三元联系：对于题干描述中，一个联系涉及到 3 个实体的情况，我们称之为三元联系。以菱形表示联系，以无向边分别连接对应的实体。

2、补充关系模式

（1）根据题干描述，找遗漏属性；

（2）题干描述已完善，根据联系归并找参照属性，通常将联系归并到实体端时，需要补充另一端实体的主键，如果联系本身存在属性，归并后也需要列出。

2、判断主键和外键

（1）找主键

（2）找外键

（3）全码：关系模型的所有属性组是这个关系模式的候选码，称为全码。

3、扩展题型

增加实体：根据题干描述添加实体，注意新实体与其他实体之间的联系。

完整性约束相关：了解主键、外键相关的概念，根据题干，做出相关判断。

规范化理论相关：了解规范化理论相关的概念，对于规范化程度没有达到 3NF 时，一般认为会存在数据冗余、修改异常、插入异常、删除异常问题。对于相关问题的解决，一般是将表进行模式分解，从而提高其规范化程度至 3NF。

(在软件设计师考试中，很少考查具体的 BCNF 规范化。)

3 章节问答

1、聚簇索引的建立是哪个阶段的任务？

答：

聚簇索引会将具体的物理存储改为顺序方式，所以属于内模式，物理设计阶段的任务。

2、数据模型的要素有哪些？

答：

数据模型的三要素：数据结构、数据操作和数据的约束条件。

3、联系类型的判断中，如果题干与日常生活经验冲突怎么处理？

答：

考试中，无论是上午题还是下午题，做题时都是以题干为主，当题干描述没有涉及时，才会依据日常惯例。如果有冲突，以题干为准。

4、SQL 语句是否需要深入学习？

答：

从考试的角度来看，SQL 在软设中的考查以 SELECT 语句为主，其他部分了解前面列举的一些函数即可。从实践的角度来看，如果从事相关工作，可以深入学习 SQL 语言。如果有时间和精力，可以自己去拓展了解。

5、关系模式如何判断其规范化程度？

答：

根据范式的判断依据，首先了解清楚一些基本概念：候选键、主属性、非主属性、部分函数依赖、传递函数依赖、函数依赖集合、函数依赖的决定因素。

然后再根据定义一步一步地深入判断：

属性不可再分则满足 1NF；

在 1NF 基础上，如果存在非主属性对候选键的部分函数依赖则当前规范化程度最高只能达到 1NF，如果已经消除了非主属性对候选键的部分函数依赖（候选键只有单属性则必定不存在对候选键的部分函数依赖），则当前规范化程度至少满足 2NF；

在 2NF 基础上，如果存在非主属性对候选键的传递函数依赖则当前规范化程度最高只能达到 2NF，如果已经消除了非主属性对候选键的传递函数依赖，则当前规范化程度至少满足 3NF（如果没有非主属性，则至少满足 3NF）；

在 3NF 基础上，如果函数依赖集合中的所有函数依赖，都满足其左侧决定因素包含候选键，则当前规范化程度至少满足 BCNF，否则最高只能达到 3NF。

6、模式分解是否保持函数依赖与有损/无损分解的判断，是否具有相关性？

答：

这是两个维度的独立判断二者没有相关性，不要混淆判断。

例 1：对于原关系模式 $R(U, F)$ ， $R=(A, B, C, D)$ ， $F=\{A \rightarrow B, C \rightarrow D\}$ ，分解为 $R_1(AB)$ ， $R_2(CD)$ ，此时保持了函数依赖但是有损。

例 2：对于原关系模式 $R(U, F)$ ， $R=(A, B, C, D)$ ， $F=\{A \rightarrow B, C \rightarrow D, A \rightarrow D\}$ ，分解为 $R_3(ABC)$ ， $R_2(CD)$ ，此时没有保持函数依赖但无损。

7、模式分解如何利用表格法判断有损/无损？

答：

表格法过程：

(1) 绘制初始表：原关系模式所有属性名作为列名，分解的关系模式作为第一行的值，根据分解后的关系模式，对已出现的属性记作 \checkmark ，未出现的属性记作 \times 。

(2) 尝试还原属性。首先查找多个关系模式中相交的属性名，接着在原函数依赖集合 F 中查看是否有以此属性为左侧决定因素的函数依赖存在，且该函数依赖在某个分解中已保留可用，根据这 3 个前提同时存在，则可以在所有存在该属性的关系模式中，将该函数依赖右侧被决定的属性还原，记作 \checkmark 。

(3) 根据当前情况，重新尝试还原属性，重复步骤 2。

(4) 直到存在一个关系模式所有属性被还原，则该关系模式无损，否则有损。

以前面的例 1、例 2 进行说明。

例 1：原关系模式 $R(U, F)$ ， $R=(A, B, C, D)$ ， $F=\{A \rightarrow B, C \rightarrow D\}$ ，分解为 $R_1(AB)$ ， $R_2(CD)$

初始表如下：

	A	B	C	D
R1 (AB)	√	√	×	×
R2 (CD)	×	×	√	√

尝试还原关系模式，不存在相交的属性，无法还原，因此该模式分解有损。

例 2：对于原关系模式 $R(U, F)$ ， $R = (A, B, C, D)$ ， $F = \{A \rightarrow B, C \rightarrow D, A \rightarrow D\}$ ，分解为 $R_3(ABC)$ ， $R_2(CD)$ 。

初始表如下：

	A	B	C	D
R1 (ABC)	√	√	√	×
R2 (CD)	×	×	√	√

尝试还原关系模式：

找到相交属性名 C，查看原函数依赖集合 F 存在 $C \rightarrow D$ ，且在分解 $R_2(CD)$ 中保留可用，3 个前提同时满足，此时可以在 R1 中还原 D 属性，得到新的表格如下：

	A	B	C	D
R1 (ABC)	√	√	√	√
R2 (CD)	×	×	√	√

R1 所有属性被还原，因此该模式分解无损。



扫码小程序获取更多【软考备考资料】