

【软考达人】

# 软考资料免费获取

- 1、最新软考题库
- 2、软考备考资料
- 3、考前压轴题



**微信扫一扫，立马获取**



**6W+ 免费题库**



**免费备考资料**

PC版题库: [ruankaodaren.com](http://ruankaodaren.com)

以下关于防火墙功能特性的叙述中，不正确的是 (1)。

- (1) A. 控制进出网络的数据包和数据流向
- B. 提供流量信息的日志和审计
- C. 隐藏内部 IP 以及网络结构细节
- D. 提供漏洞扫描功能

【答案】D

【解析】防火墙是被动防御，无法提供系统漏洞扫描

在程序执行过程中，Cache 与主存的地址映射是由 (2) 完成的。

- (2) A. 操作系统
- B. 程序员调度
- C. 硬件自动
- D. 用户软件

【答案】C

【解析】典型的存储体系结构分为“高速缓冲存储器(Cache)←→主存←→辅存”三个层次。地址映像的作用之一是将 CPU 送来的主存地址转换成 Cache 地址。在程序的执行过程中，Cache 与主存的地址映像由专门的硬件电路自动完成。

某四级指令流水线分别完成取指、取数、运算、保存结果四步操作。若完成上述操作的时间依次为 8ns、9ns、4ns、8ns，则该流水线的操作周期应至少为 (3) ns。

- (3) A. 4
- B. 8
- C. 9
- D. 33

【答案】C

【解析】流水线稳定之后，9s 就会完成一个周期

内存按字节编址。若用存储容量为 32Kx8bit 的存储器芯片构成地址从 A0000H 到 DFFFFH 的内存，则至少需要 (4) 片芯片。

- (4) A. 4
- B. 8
- C. 16
- D. 32

【答案】B

【解析】 $DFFFFH - A0000H + 1 = 40000H = 4 \times 16^4$ ；因为是按字节编址，所以所需芯片数为： $4 \times 16^4 / (32 \times 1024) = 8$

计算机系统的主存主要是由 (5) 构成的。

- (5) A. DRAM
- B. SRAM
- C. Cache
- D. EEPROM

【答案】A

【解析】

DRAM (Dynamic Random Access Memory)，即动态随机存取存储器，最为常见的系统内存。

SRAM (Static Random Access Memory)，即静态随机存取存储器。它是一种具有静止存取功能的内存，不需要刷新电路即能保存它内部存储的数据。

Caché，是一种后关系型数据库。能并发访问同一数据的数据库技术。

EEPROM (Electrically Erasable Programmable read only memory)，电可擦可编程只读存储器--一种掉电后数据不丢失的存储芯片。EEPROM 可以在电脑上或专用设备上擦除已有信息，重新编程。一般用在即插即用。

以下关于海明码的叙述中，正确的是(6)。

- (6) A. 海明码利用奇偶性进行检错和纠错  
B. 海明码的码距为 1  
C. 海明码可以检错但不能纠错  
D. 海明码中数据位的长度与校验位的长度必须相同

【答案】A

【解析】海明码利用了奇偶校验位的概念，通过在数据位后面增加一些比特，可以验证数据的有效性。利用一个以上的校验位，汉明码不仅可以验证数据是否有效，还能在数据出错的情况下指明错误位置。 $2^P \geq P+D+1$ ，其中 P 代表汉明码的个数，D 代表数据位的个数。

计算机运行过程中，CPU 需要与外设进行数据交换。采用(7)控制技术时，CPU 与外设可并行工作。

- (7) A. 程序查询方式和中断方式  
B. 中断方式和 DMA 方式  
C. 程序查询方式和 DMA 方式  
D. 程序查询方式、中断方式和 DMA 方式

【答案】B

【解析】

程序查询方式是通过 CPU 执行程序来查询状态的。

DMA(Direct Memory Access，直接内存存取) 是所有现代电脑的重要特色，它允许不同速度的硬件装置来沟通，而不需要依赖于 CPU 的大量中断负载。否则，CPU 需要从来源把每一片段的资料复制到暂存器，然后把它们再次写回到新的地方。在这个时间中，CPU 对于其他的工作来说就无法使用。

有 DMA 无程序查询

与 HTTP 相比，HTTPS 协议对传输的内容进行加密，更加安全。HTTPS 基于 (8) 安全协议，其默认端口是 (9)。

- (8) A. RSA                      B. DES                      C. SSL                      D. SSH  
(9) A. 1023                      B. 443                      C. 80                      D. 8080

【答案】C    B

【解析】HTTPS（全称：Hyper Text Transfer Protocol over Secure Socket Layer），是以安全为目标的 HTTP 通道，简单讲是 HTTP 的安全版。即 HTTP 下加入 SSL 层，HTTPS 的安全基础是 SSL，因此加密的详细内容就需要 SSL。它是一个 URI scheme（抽象标识符体系），句法类同 http:体系。HTTPS 使用端口 443，而不是像 HTTP 那样使用端口 80 来和 TCP/IP 进行通信。

下列攻击行为中，属于典型被动攻击的是 (10)。

- (10) A. 拒绝服务攻击                      B. 会话拦截  
C. 系统干涉                      D. 修改数据命令

【答案】C

【解析】被动攻击主要是收集信息而不是进行访问，数据的合法用户对这种活动一点也不会觉察到。被动攻击包括嗅探、信息收集等攻击方法。攻击方不知道被攻击方什么时候对话

(11) 不属于入侵检测技术。

- (11) A. 专家系统                      B. 模型检测                      C. 简单匹配                      D. 漏洞扫描

【答案】D

【解析】

漏洞扫描为另一种安全防护策略。

某软件公司项目组的程序员在程序编写完成后均按公司规定撰写文档，并上交公司存档。

此情形下，该软件文档著作权应由 (12) 享有。

- (12) A. 程序员                      B. 公司与项目组共同  
C. 公司                      D. 项目组全体人员

【答案】C

【解析】

属于职务作品。

我国商标法规定了申请注册的商标不得使用的文字和图形，其中包括县级以上行政区的地名(文字)。以下商标注册申请，经审查，能获准注册的商标是(13)。

- (13) A. 青岛(市)      B. 黄山(市)      C. 海口(市)      D. 长沙(市)

【答案】A

【解析】青岛啤酒记着没，不是打广告，不过我考试的时候第一反应就是这个。

李某购买了一张有注册商标的应用软件光盘，则李某享有(14)。

- (14) A. 注册商标专用权      B. 该光盘的所有权  
C. 该软件的著作权      D. 该软件的所有权

【答案】B

【解析】

某医院预约系统的部分需求为：患者可以查看医院发布的专家特长介绍及其就诊时间；系统记录患者信息，患者预约特定时间就诊。用 DFD 对其进行功能建模时，患者是(15)；用 ERD 对其进行数据建模时，患者是(16)。

- (15) A. 外部实体      B. 加工      C. 数据流      D. 数据存储  
(16) A. 实体      B. 属性      C. 联系      D. 弱实体

【答案】A A

【解析】数据流图(Data Flow Diagram)：简称 DFD，它从数据传递和加工角度，以图形方式来表达系统的逻辑功能、数据在系统内部的逻辑流向和逻辑变换过程，是结构化系统分析方法的主要表达工具及用于表示软件模型的一种图示方法。

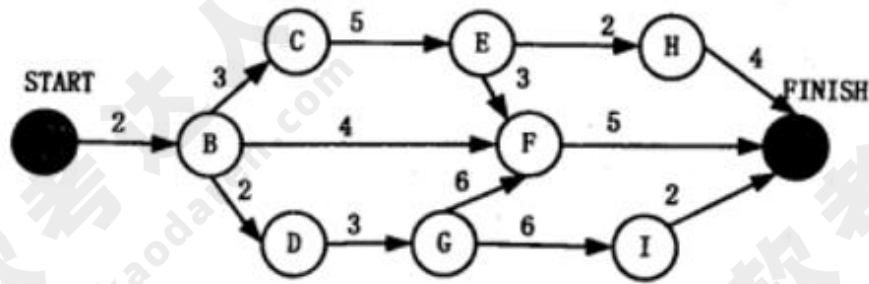
E-R 图也称实体-联系图(Entity Relationship Diagram)，提供了表示实体类型、属性和联系的方法，用来描述现实世界的概念模型。在实际情况中并非总有一个属性是键，为描述这类情况，E-R 图引入了弱实体元素。即没有键属性的实体型。例如本地电话号码、寝室等（如果其仅按照顺序编号的话），对应地，存在键属性的实体型则为强实体型。

不管是哪种建模方式患者都是实体

某软件项目的活动图如下图所示，其中顶点表示项目里程碑，链接顶点的边表示包含的活动，变色数字表示活动的持续时间（天）。完成该项目的最少时间为(17)天。由于某种



原因，现在需要同一个开发人员完成 BC 和 BD，则完成该项目的最少时间为 (18) 天。



(17) A. 11 B. 18 C. 20 D. 21

(18) A. 11 B. 18 C. 20 D. 21

【答案】B D

【解析】

(17) 关键路径为 ABCEFI 和 ABDGFI 18 天

(18) BC 持续时间 3 天，BD 持续时间 2 天，由一天完成，则可以把 BC 持续时间作为 5 天，BD 持续时间也为 5 天，则关键路径为 ABDGFI，21 天

某企业财务系统的需求中，属于功能需求的是 (19)。

- (19) A. 每个月特定的时间发放员工工资  
B. 系统的响应时间不超过 3 秒  
C. 系统的计算精度符合财务规则的要求  
D. 系统可以允许 100 个用户同事查询自己的工资

【答案】A

【解析】功能需求：系统要做什么，在何时做，在何时以及如何修改升级。

更适合用来开发操作系统的编程语言是 (20)。

- (20) A. C/C++ B. Java C. Python D. JavaScript

【答案】A

【解析】Linux 就是用 C 开发的

以下关于程序设计语言的叙述中，不正确的是 (21)。

- (21) A. 脚本语言中不使用变量和函数

- B. 标记语言常用于描述格式化和链接
- C. 脚本语言采用解释方式实现
- D. 编译型语言的执行效率更高

【答案】A

【解析】脚本语言中可以使用变量

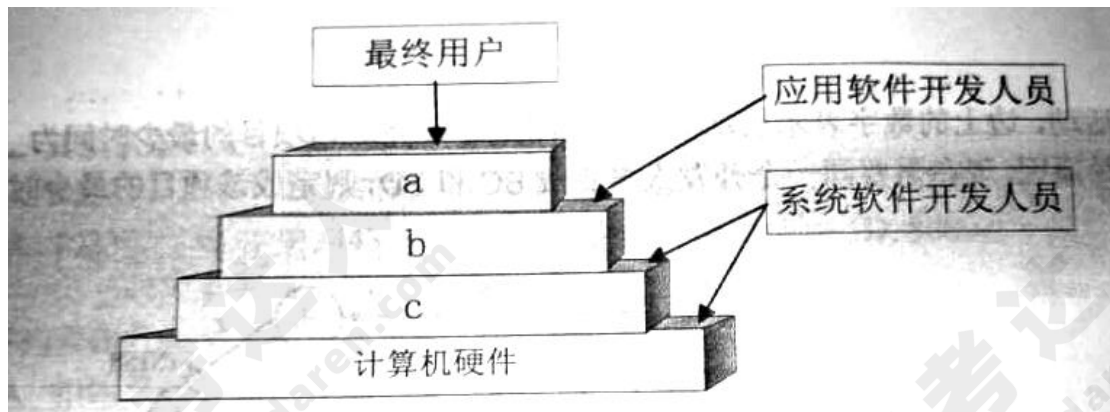
将高级语言源程序通过编译或解释方式进行翻译时，可以先生成与源程序等价的某种中间代码。以下关于中间代码的叙述中，正确的是 (22)。

- (22) A. 中间代码常采用符号表来表示
- B. 后缀式和三地址码是常用的中间代码
  - C. 对中间代码进行优化要依据运行程序的机器特性
  - D. 中间代码不能跨平台

【答案】B

【解析】中间代码常用的表达形式如后缀式，三元式，四元式和树，可以跨平台

计算机系统的层次结构如下圈所示，基于硬件之上的软件可分为 a、b 和 c 三个层次。图中 a、b 和 c 分别表示 (23)。



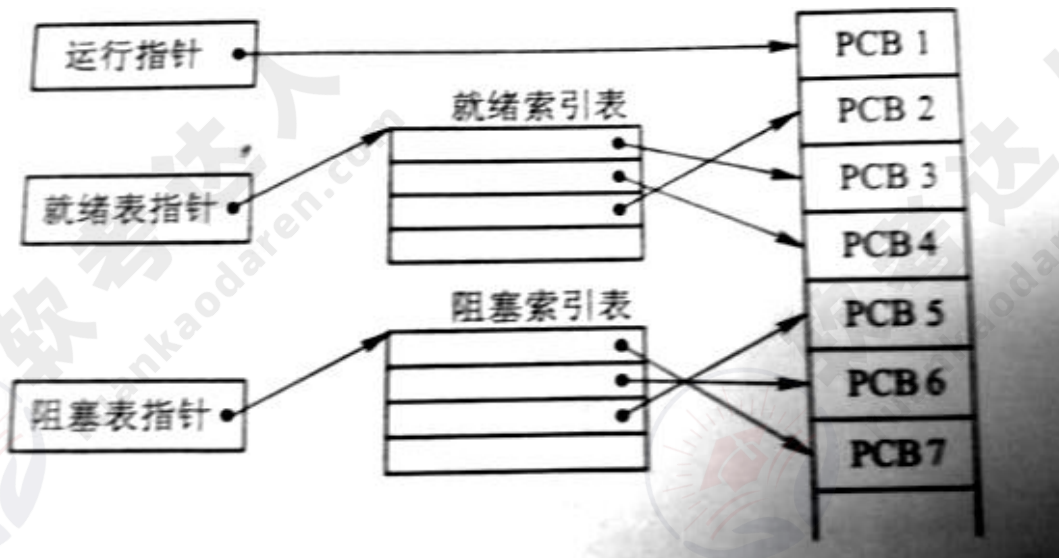
- (23) A. 操作系统、系统软件和应用软件
- B. 操作系统、应用软件和系统软件
- C. 应用软件、系统软件和操作系统
- D. 应用软件、操作系统和系统软件

【答案】C

【解析】

系统软件开发人员开发操作系统和系统软件。应用软件开发人员开发应用软件。

下图所示的 PCB（进程控制块）的组织方式是 (24)，图中 (25)。



(24) A. 链接方式                      B. 索引方式                      C. 顺序方式                      D. Hash

(25) A. 有 1 个运行进程、2 个就绪进程、4 个阻塞进程  
 B. 有 1 个运行进程、2 个就绪进程、4 个阻塞进程  
 C. 有 1 个运行进程、3 个就绪进程、3 个阻塞进程  
 D. 有 1 个运行进程、4 个就绪进程、2 个阻塞进程

【答案】B C

【解析】都给出索引表了，这就不用说了；运行指针就指向了一个进程，就绪有三个，阻塞有三个。

某文件系统采用多级索引结构。若磁盘块的大小为 1K 字节，每个块号占 3 字节，那么采用二级索引时的文件最大长度为 (26) 字节。

(26) A. 1024                      B. 2048                      C. 116281                      D. 232562

【答案】C

【解析】

某操作系统采用分页存储管理方式，下图给出了进程 A 和进程 B 的页表结构。如果物理页的大小为 1K 字节，那么进程 A 中逻辑地址为 1024（十进制）用变量存放在 (27) 号物理内存页中。假设进程 A 的逻辑页 4 与进程 B 的逻辑页 5 要共享物理页 4，那么应该在进程 A 页表的逻辑页 4 和进程 B 页表的逻辑页 5 对应的物理页处分别填 (28)。



进程 A 页表		进程 B 页表		物理页
逻辑页	物理页	逻辑页	物理页	
0	8	0	1	0
1	3	1	6	1
2	5	2	9	2
3	2	3	7	3
4		4	0	4
5		5		5
				6
				7
				8
				9

(27) A. 8

B. 3

C. 5

D. 2

(28) A. 4、4

B. 4、5

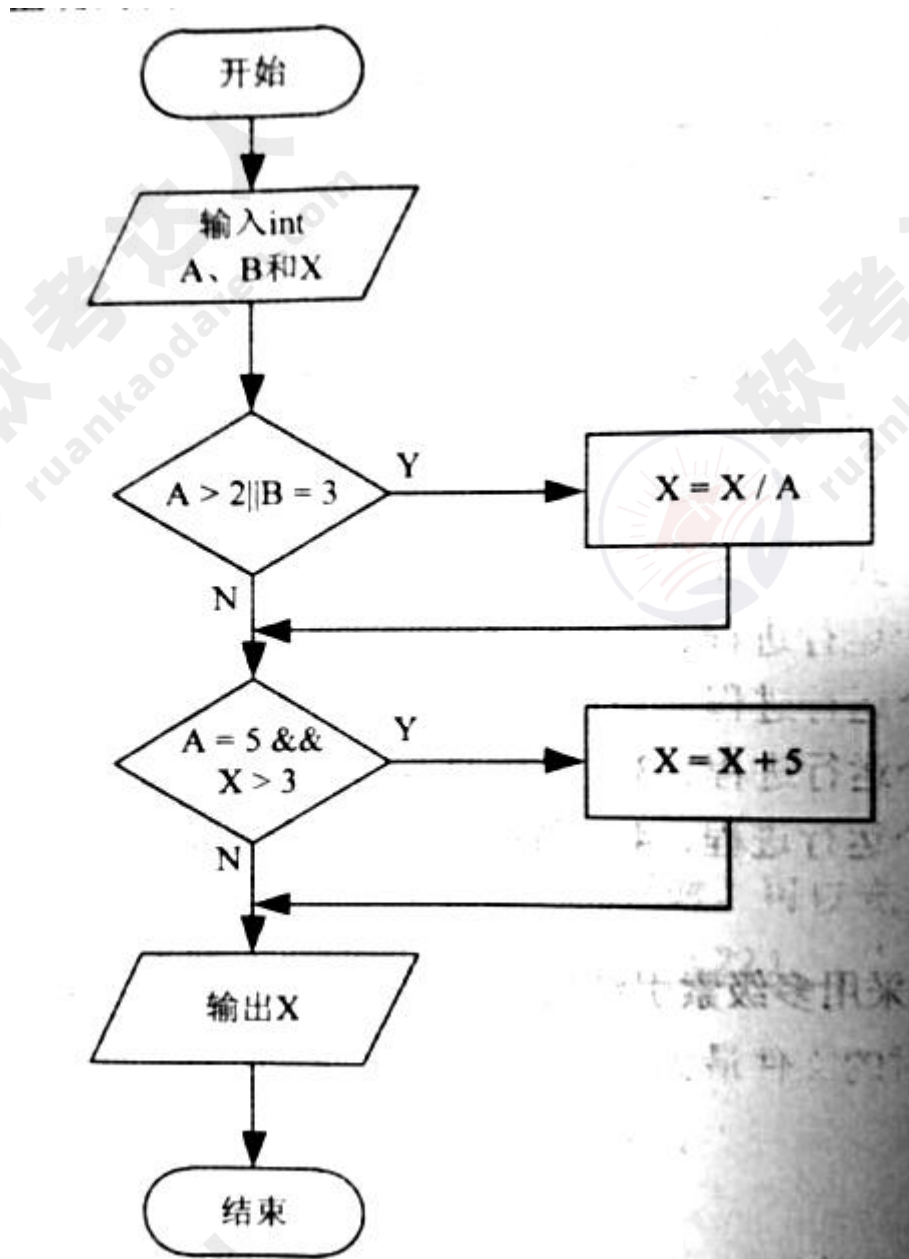
C. 5、4

D. 5、5

【答案】B A

【解析】 $1k=1*1024$ ；所以逻辑页为 1，对应物理页为 3；第二空是可以共用的。

用自盒测试方法对如下图所示的流程图进行测试。若要满足分支覆盖，则至少要 (29) 个测试用例，正确的测试用例对是 (30) (测试用例的格式为 (A, B, X; X))。



- (29) A. 1                      B. 2                      C. 3                      D. 4
- (30) A. (1, 3, 3; 3) 和 (5, 2, 15; 3)                      B. (1, 1, 5; 5) 和 (5, 2, 20; 9)
- C. (2, 3, 10; 5) 和 (5, 2, 18; 3)                      D. (5, 2, 16; 3) 和 (5, 2, 21; 9)

【答案】B B

【解析】第二空已经给出第一空的答案两个；第二空用代入法。

配置管理贯穿软件开发的整个过程。以下内容中，不属于配置管理的是(31)。

- (31) A. 版本控制                      B. 风险管理                      C. 变更管理                      D. 配置状态报告

【答案】B

【解析】风险管理属于风险控制。

极限编程（XP）的十二个最佳实践不包括（32）。

- (32) A. 小的发布                      B. 结对编程                      C. 持续集成                      D. 精心设计

【答案】D

【解析】极限编程（XP）

4 大价值观：沟通，简单性，简单，反馈与勇气；

5 个原则：快速反馈，简单性假设，逐步修改，提倡更改和优质工作

12 个最佳实践：计划游戏（快速制定计划、随着细节的不断变化而完善）、小型发布（系统的设计要能够尽可能早地交付）、隐喻（找到合适的比喻传达信息）、简单设计（只处理当前的需求，使设计保持简单）、测试现行（先写测试代码，然后再编写程序）、重构（重新审视需求和设计，重新明确地描述它们以符合新的和现有的需求）、结对编程、集体代码所有制、持续集成（可以按日甚至按小时为客户提供可运行的版本）、每周工作 40 个小时、现场客户和编码标准。

以下关于管道过滤器体系结构的有点的叙述中，不正确的是（33）。

- (33) A. 软件构件具有良好的高内聚、低耦合的特点  
B. 支持重用  
C. 支持并行执行  
D. 提高性能

【答案】D

【解析】管道-过滤器模式的体系结构是面向数据流的软件体系结构。它最典型的应用是在编译系统。一个普通的编译系统包括词法分析器,语法分析器,语义分析与中间代码生成器,优化器,目标代码生成器等一系列对源程序进行处理的过程。人们可以将编译系统看作一系列过滤器的连接体,按照管道-过滤器的体系结构进行设计。此外,这种体系结构在其它一些领域也有广泛的应用。因此它成为软件工程和软件开发中的一个突出的研究领域。

模块 A 将学生信息，即学生姓名、学号、手机号等放到一个结构体中，传递给模块 B。

模块 A 和 B 之间的耦合类型为（34）耦合。

- (34) A. 数据                      B. 标记                      C. 控制                      D. 内容

【答案】A

【解析】功能内聚：完成一个单一功能，各个部分协同工作，缺一不可。

顺序内聚：处理元素相关，而且必须顺序执行。

通信内聚：所有处理元素集中在一个数据结构的区域上。

过程内聚：处理元素相关，而且必须按特定的次序执行。

瞬时内聚：所包含的任务必须在同一时间间隔内执行（如初始化模块）。

逻辑内聚：完成逻辑上相关的一组任务。

偶然内聚：完成一组没有关系或松散关系的任务。

某模块内涉及多个功能，这些功能必须以特定的次序执行，则该模块的内聚类型为(35)

内聚。

(35)A. 实践

B. 过程

C. 信息

D. 功能

【答案】B

【解析】

系统交付用户使用后，为了改进系统的图形输出而对系统进行修改的维护行为属于(36)

维护。

(36)A. 改正性

B. 适应性

C. 改善性

D. 预防性

【答案】C

【解析】正确性维护：改正没有发现的错误

适应性维护：技术，需求发生了变化而引起的维护

完善性维护：扩展功能和完善性能

预防性维护：提高可靠性和可维护性

在面向对象方法中，将逻辑上相关的数据以及行为绑定在一起，使信息对使用者隐蔽称为(37)。当类中的属性或方法被设计为 private 时，(38) 可以对其进行访问。

(37)A. 抽象

B. 继承

C. 封装

D. 多态

(38)A. 应用程序中所有方法

B. 只有此类中定义的方法

C. 只有此类中定义的 public 方法

D. 同一个包中的类中定义的方法

【答案】C B

【解析】

封装是指利用抽象数据类型将数据和基于数据的操作封装在一起,使其构成一个不可分割的独立实体,数据被保护在抽象数据类型的内部,尽可能地隐藏内部的细节,只保留一些对外接口使之与外部发生联系。

只有此类中定义的方法可以对私有成员进行访问。

采用继承机制创建子类时,子类中\_(39)。

- (39) A. 只能有父类中的属性                      B. 只能有父类中的行为  
C. 只能新增行为                      D. 可以有新的属性和行为

【答案】D

【解析】

子类相对于父类,要更加特殊。所以会有新的成员来描述其特殊。

面向对象分析过程中,从给定需求描述中选择\_(40)来识别对象。

- (40) A. 动词短语              B. 名词短语              C. 形容词              D. 副词

【答案】B

【解析】

名词短语暗示类及其属性动词和动词短语暗示对职责或操作。

如图所示的UML类图中,Shop和Magazine之间为\_(41)关系,Magazine和Page之间为\_(42)关系。UML类图统称不用于对\_(43)进行建模。



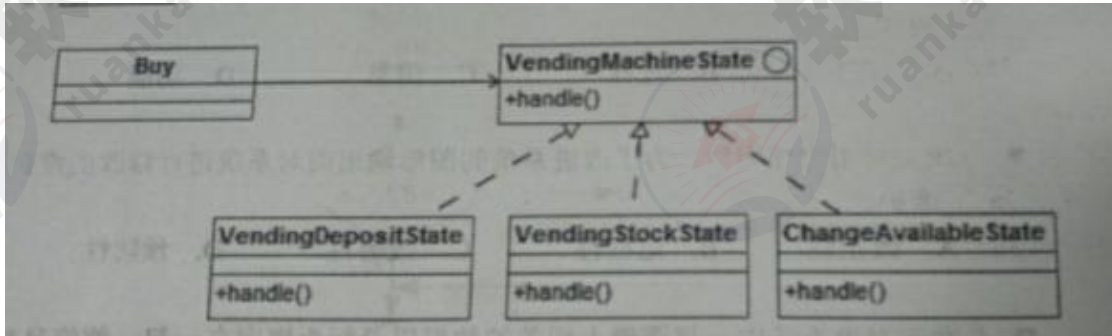
- (41) A. 关联                      B. 依赖                      C. 组合                      D. 继承  
(42) A. 关联                      B. 依赖                      C. 组合                      D. 继承  
(43) A. 系统的词汇                      B. 简单的协作  
C. 逻辑数据库模式                      D. 对象快照

【答案】A C D

【解析】第一个空和第二个空,可以通过他们之间的连线得出;第三空:对系统的词汇建模;对简单的协作建模;对逻辑数据库模式建模都可以用到类。



自动售货机根据库存、存放货币量、找零能力、所选项目等不同，在货币存入并进行选择时具有如下行为：交付产品不找零：交付产品找零：存入货币不足而不提供任何产品；库存不足而不提供任何产品。这一业务需求适合采用(44)模式设计实现，其类图如下图所示，其中(45)是客户程序使用的主要接口，可用状态来对其进行配置。此模式为(46)，体现的最主要的意图是(47)。



- (44) A. 观察者 (Observer)                      B. 状态 (State)  
       C. 策略 (Strategy)                      D. 访问者 (Visitor)
- (45) A. VendingMachineState                      B. Buy  
       C. VendingDepositState                      D. VendingStockState
- (46) A. 创建型对象模式                      B. 结构型对象模式  
       C. 行为型类模式                      D. 行为型对象模式
- (47) A. 当一个对象状态改变时所有依赖它的对象得到通知并自动更新  
       B. 在不破坏封装性的前提下，捕获对象的内部状态并在对象之外保存  
       C. 一个对象在其内部状态改变时改变其行为  
       D. 将请求封装为对象从而可以使用不同的请求对客户进行参数化

【答案】B A D C

【解析】

有 State 关键词 状态模式：允许一个对象在其内部状态改变时改变它的行为。对象看起来似乎修改了它的类。在很多情况下，一个对象的行为取决于一个或多个动态变化的属性，这样的属性叫做状态，这样的对象叫做有状态的 (stateful) 对象，这样的对象状态是从事先定义好的一系列值中取出的。当一个这样的对象与外部事件产生互动时，其内部状态就会改变，从而使得系统的行为也随之发生变化。

是整个状态模式的基类也是对外的接口

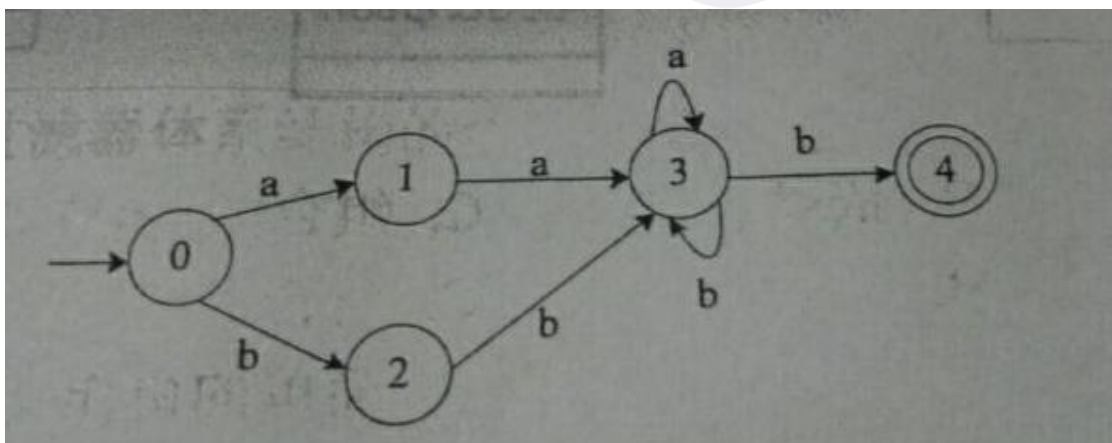
编译过程中进行的语法分析主要是分析 (48)。

- (48) A. 源程序中的标识符是否合法      B. 程序语句的含义是否合法  
C. 程序语句的结构是否合法      D. 表达式的类型是否合法

【答案】D

【解析】语法分析的任务是根据语言的语法规则分析单词串是否构成短语和句子，即表达式，语句和程序等基本语言结构，同时检查和处理程序中的语法错误。

某确定的有限自动机 (DFA) 的状态转换图如下图所示 (0 是初态，4 是终态)，则该 DFA 能识别 (49)。

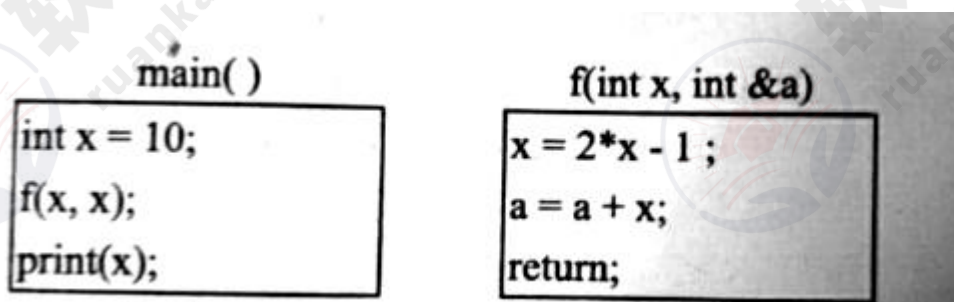


- (49) A. aaab      B. abab      C. bbba      D. abba

【答案】A

【解析】一定是以 b 结束，aa...b 或 bb...b;

函数 main ()、f () 的定义如下所示。调用函数 f () 时，第一个参数采用传值 (call by value) 方式，第二个参数采用传引用 (call by referen) 方式，则函数 main () 执行后输出的值为 (50)。



(50) A. 10

B. 19

C. 20

D. 29

【答案】D

【解析】

 $x = 2 * 10 - 1 = 19$   $a = a + x = 10 + 19 = 29$   $a$  是引用传递 所以结果为 29

采用三级结构/两级映像的数据库体系结构，如果对数据库的一张表创建聚簇索引，改变的是数据库的 (51)。

(51) A. 用户模式

B. 外模式

C. 模式

D. 内模式

【答案】B

【解析】索引对应外模式；视图对应概念模式；基本表对应内模式

某企业的培训关系模式  $R(\text{培训科目}, \text{培训师}, \text{学生}, \text{成绩}, \text{时间}, \text{教室})$ ， $R$  的函数依赖集  $F = \{\text{培训科目} \rightarrow \text{培训师}, (\text{学生}, \text{培训科目}) \rightarrow \text{成绩}, (\text{时间}, \text{教室}) \rightarrow \text{培训科目}, (\text{时间}, \text{培训师}) \rightarrow \text{教室}, (\text{时间}, \text{学生}) \rightarrow \text{教室}\}$ 。关系模式  $R$  的主键为 (52)，其规范化程度最高达到 (53)。

(52) A. (学生, 培训科目)

B. (时间, 教室)

C. (时间, 培训师)

D. (时间, 学生)

(53) A. 1NF

B. 2NF

C. 3NF

D. BCNF

【答案】D B

【解析】第一空：做这类图先把依赖其它对象的去掉，不够的话再添加；第二空：属性不可再分，没有部分依赖，但是有传递依赖，(时间, 学生) 可以推出培训师。

设关系模式  $R(U, F)$ ，其中： $U = \{A, B, C, D, E\}$ ， $F = \{A \rightarrow B, DE \rightarrow B, CB \rightarrow E, E \rightarrow A, B \rightarrow D\}$ 。(54) 为关系模式  $R$  的候选关键字。分解 (55) 是无损连接，并保持函数依赖的。

(54) A. AB

B. DE

C. DB

D. CE

(55) A.  $\rho = \{R_1(AC), R_2(ED), R_3(B)\}$ B.  $\rho = \{R_1(AC), R_2(E), R_3(DB)\}$ C.  $\rho = \{R_1(AC), R_2(ED), R_3(AB)\}$ D.  $\rho = \{R_1(ABC), R_2(ED), R_3(ACE)\}$

【答案】D D

【解析】参考 52；排除法，有选项中推导出的依赖关系不变，没有丢失元素

在基于 Web 的电子商务应用中，访问存储于数据库中的业务对象的常用方式之一是(56)。

- (56) A. JDBC B. XML C. CGI D. COM

【答案】A

【解析】JDBC：Java 数据库连接模式

设 S 是一个长度为 n 的非空字符串，其中的字符各不相同，则其互异的非平凡子串（非空且不同于 S 本身）个数为(57)。

- (57) A.  $2n-1$  B.  $n^2$  C.  $n(n+1)/2$  D.  $(n+2)(n-1)/2$

【答案】D

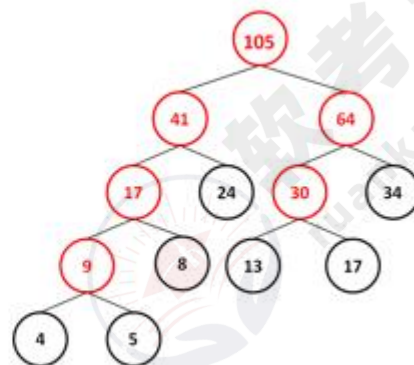
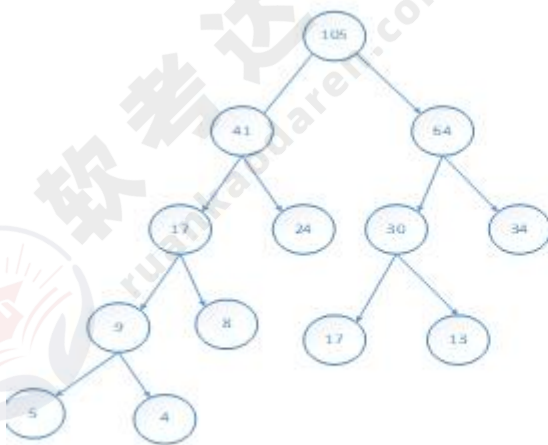
【解析】假设  $n=1,2,3$  排除法

假设某消息中只包含 7 个字符  $\{a, b, c, d, e, f, g\}$ ，若 7 个字符在消息中出现的次数为  $\{5, 24, 8, 17, 34, 13, 1\}$ ，利用哈夫曼树（最优二叉树）为该消息中的字符构造符合前缀编码要求的不等长编码。各字符的编码长度分别为(58)。

- (58) A. a:4, b:2, c:3, d:3, e:2, f:4, g:3 B. a:6, b:2, c:5, d:3, e:1, f:6, g:4  
C. a:3, b:3, c:3, d:3, e:3, f:2, g:3 D. a:2, b:6, c:3, d:5, e:6, f:1, g:4

【答案】A

【解析】



设某二叉树采用二叉链表表示（即结点的两个指针分别指示左、右孩子）。当该二叉树包含  $k$  个节点时，其二叉链表节点中必有 (59) 个空的汉子指针。

- (59) A.  $k-1$       B.  $k$       C.  $k+1$       D.  $2k$

【答案】C

【解析】

二叉树的的二叉链表存储结构中每个结点有 2 个指针。每个结点有 0 个、1 个或者 2 个空指针对应 2 个、1 个、0 个非空指针。

二叉树中边的个数等于非空指针的个数。

假设二叉树中节点的总个数为  $N$ ,

假设二叉树中边的个数为  $M$

假设二叉树中度为 0 的结点的个数为  $n_0$ ,

假设二叉树中度为 1 的结点的个数为  $n_1$ ,

假设二叉树中度为 2 的结点的个数为  $n_2$ .

所以有  $n_0+n_1+n_2=N$  ----- (1)

二叉树中除了根结点之外，其他的结点都有一条边进入该结点，所以二叉树中边的总个数为

$M=N-1$ ;----- (2)

又  $M=n_1+2*n_2$ ;----- (3)

所以由 (1) (2) (3) 可得  $n_0=n_2+1$ ;----- (4)

设空节点的 个数为  $K$  , 则  $K=2*n_0+n_1$ ----- (5)

结合 (1) (4) (5) 可以得到  $K=N+1$ . (空指针的的个数比结点总个数多 1)

由 (2) 可以知道 边数  $M=N-1$ ; (二叉树的边数为结点个数减 1)

由 (4) 可以知道度为 0 的结点的个数 (叶子结点个数) = 度为 2 的结点个数 + 1 ( $n_0=n_2+1$ ;) )

以下关于无向连通图  $G$  的叙述中，不正确的是 (60)。

- (60) A.  $G$  中任意两个顶点之间均有边存在  
 B.  $G$  中任意两个顶点之间存在路径  
 C. 从  $G$  中任意顶点出发可遍历图中所有顶点  
 D.  $G$  的邻接矩阵是对称矩阵

【答案】A



【解析】在连通图中可能有两个顶点不存在直接相连的边。

两个递增序列 A 和 B 的长度分别为 m 和 n ( $m < n$  且 m 与 n 接近)，将二者归并为一个长度为  $m+n$  的递增序列。当元素关系为 (61)，归并过程中元素的比较次数最少。

- (61) A.  $a_1 < a_2 < \dots < a_{m-1} < a_m < b_1 < b_2 < \dots < b_{n-1} < b_n$   
 B.  $b_1 < b_2 < \dots < b_{n-1} < b_n < a_1 < a_2 < \dots < a_{m-1} < a_m$   
 C.  $a_1 < b_1 < a_2 < b_2 < \dots < a_{m-1} < b_{m-1} < a_m < b_m < b_{m+1} < \dots < b_{n-1} < b_n$   
 D.  $b_1 < b_2 < \dots < b_{m-1} < b_m < a_1 < a_2 < \dots < a_{m-1} < a_m < b_{m+1} < \dots < b_{n-1} < b_n$

【答案】A

【解析】

若 A 的最大元素小于 B 的最小元素，则只需要比较 m 次，这时归并过程中元素的比较次数最少。

求解两个长度为 n 的序列 X 和 Y 的一个最长公共序列（如序列 ABCBDAB 和 BDCABA 的一个最长公共子序列为 BCBA）可以采用多种计算方法。如可以采用蛮力法，对 X 的每一个子序列，判断其是否也是 Y 的子序列，最后求出最长的即可，该方法的时间复杂度为 (62)。经分析发现该问题具有最优子序列，可以定义序列成都分别为 i 和 j 的两个序列 X 和 Y 的最长公共子序列的成都为  $C[i, j]$ ，如下式所示。

$$C[i, j] = \begin{cases} 0 & \text{若 } i=0 \text{ 或 } j=0 \\ C[i-1, j-1] + 1 & \text{若 } i, j > 0 \text{ 且 } x_i = y_j \\ \max(C[i-1, j], C[i, j-1]) & \text{其它} \end{cases}$$

(62) A.  $O(n^2)$       B.  $O(n^2 \lg n)$       C.  $O(n^3)$       D.  $O(n^2)$

(63) A.  $O(n^2)$       B.  $O(n^2 \lg n)$       C.  $O(n^3)$       D.  $O(n^2)$

【答案】D A

【解析】第二个空是动态规划法时间复杂度为  $O(n)$

现需要对一个基本有序的数组进行排序。此时最适宜采用的算法为 (64) 排算法，时间复杂度为 (65)。

- (64) A. 插入 B. 快速 C. 归并 D. 堆  
(65) A.  $O(n)$  B.  $O(n \lg n)$  C.  $O(n^2)$  D.  $O(n^2 \lg n)$

【答案】A A

【解析】插入排序对基本有序的数组排序速度快；插入排序的时间复杂度都是  $O(n^2)$

相比于 TCP，UDP 的优势为 (66)。

- (66) A. 可靠传输 B. 开销较小 C. 拥塞控制 D. 流量控制

【答案】B

【解析】

TCP 与 UDP 区别：1、TCP 面向连接（如打电话要先拨号建立连接）；UDP 是无连接的，即发送数据之前不需要建立连接；2、TCP 提供可靠的服务。也就是说，通过 TCP 连接传送的数据，无差错，不丢失，不重复，且按序到达；UDP 尽最大努力交付，即不保证可靠交付；3、TCP 面向字节流，实际上是 TCP 把数据看成一连串无结构的字节流；UDP 是面向报文的；UDP 没有拥塞控制，因此网络出现拥塞不会使源主机的发送速率降低（对实时应用很有用，如 IP 电话，实时视频会议等）；4、每一条 TCP 连接只能是点到点的；UDP 支持一对一，一对多，多对一和多对多的交互通信；5、TCP 首部开销 20 字节；UDP 的首部开销小，只有 8 个字节；6、TCP 的逻辑通信信道是全双工的可靠信道，UDP 则是不可靠信道，整体来看 UDP 开销较小。

若一台服务器只开放了 25 和 110 两个端口，那么这台服务器可以提供 (67) 服务。

- (67) A. E-Mail B. WEB C. DNS D. FTP

【答案】A

【解析】25，110 号端口都与邮件的相关协议有关

SNMP 是一种异步请求/响应协议，采用 (68) 协议进行封装。

- (68) A. IP B. ICMP C. TCP D. UDP

【答案】D

【解析】SNMP：简单网络管理协议

在一台安装好 TCP/IP 协议的计算机上，当网络连接不可用时，为了测试编写好的网络

程序，通常使用的目的主机 IP 地址为 (69)。

- (69) A. 0.0.0.0      B. 127.0.0.1      C. 10.0.0.1      D. 210.225.21.255/24

【答案】B

【解析】127.0.0.1 是主机用来做环回测试的

测试网络连通性通常采用的命令是 (70)。

- (70) A. Nstar      B. Ping      C. Mscinfug      D. Cmd

【答案】B

【解析】ping 命令还有 trace 命令是用来测试网络连通性的。

The development of the Semantic Web proceeds in steps, each step building a layer on top of another. The pragmatic justification for this approach is that it is easier to achieve (71) on small steps, whereas it is much harder to get everyone on board if too much is attempted. Usually there are several research groups moving in different directions; this (72) of ideas is a major driving force for scientific progress. However, from an engineering perspective there is a need to standardize. So, if most researchers agree on certain issues and disagree on others, it makes sense to fix the point of agreement. This way, even if the more ambitious research efforts should fail, there will be at least (73) positive outcomes.

Once a (74) has been established, many more groups and companies will adopt it, instead of waiting to see which of the alternative research lines will be successful in the end. The nature of the Semantic Web is such that companies and single users must build tools, add content, and use that content. We cannot wait until the full Semantic Web vision materializes—it may take another ten years for it to be realized to its full (75) (as envisioned today, of course).

- (71) A. conflicts      B. consensus      C. success      D. disagreement  
 (72) A. competition      B. agreement      C. cooperation      D. collaboration  
 (73) A. total      B. complete      C. partial      D. entire  
 (74) A. technology      B. standard      C. pattern      D. model  
 (75) A. area      B. goal      C. object      D. extent

【答案】C A C A D

【解析】

语义网络的发展是一步一步的，每一步都建立在另一部的基础之上。选择这种方法的现实理由就是因为很容易对一小步取得成功，而如果想要一步到位就难得多。通常，很多研究组织都是从不同方向考虑的，这种思想的竞争的方式是科学进步的驱动力。然而，从工程的角度来说是需要进行标准化的。因此，如果大多数研究者同意某个观点不同意另一个的时候，改正观点是有意义的。这样，即使再宏大的研究努力也会失败，可能会有局部的积极效果。一旦一个技术被建立，许多组织和企业都会采纳，而不是等待并查看其他研究线是否会获得成果。语义网络的性质就是让企业和单个用户必须构建工具，添加内容并使用。我们不会等着整个语义网络被物化。因为实现它的全部内容需要再过十年时间（当然是按照今天所设想）。

## 试题一

阅读下列说明和图，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

### 【说明】

某公司拟开发一个共享单车系统，采用北斗定位系统进行单车定位，提供针对用户的 APP 以及微信小程序、基于 Web 的管理与监控系统。该共享单车系统的主要功能如下。

1) 用户注册登录。用户在 APP 端输入手机号并获取验证码后进行注册，将用户信息进行存储。用户登录后显示用户所在位置周围的单车。

2) 使用单车。

①扫码/手动开锁。通过扫描二维码或手动输入编码获取开锁密码，系统发送开锁指令进行开锁，系统修改单车状态，新建单车行程。

②骑行单车。单车定时上传位置，更新行程。

③锁车结账。用户停止使用或手动锁车并结束行程后，系统根据已设置好的计费规则及使用时间自动结算，更新本次骑行的费用并显示给用户，用户确认支付后，记录行程的支付状态。系统还将重置单车的开锁密码和单车状态。

3) 辅助管理。

①查询。用户可以查看行程列表和行程详细信息。

②报修。用户上报所在位置或单车位置以及单车故障信息并进行记录。

4) 管理与监控。

①单车管理及计费规则设置。商家对单车基础信息、状态等进行管理，对计费规则进行设置并存储。

②单车监控。对单车、故障、行程等进行查询统计。

③用户管理。管理用户信用与状态信息，对用户进行查询统计。现采用结构化方法对共享单车系统进行分析与设计，获得如图 1-1 所示的上下文数据流图和图 1-2 所示的 0 层数据流图。



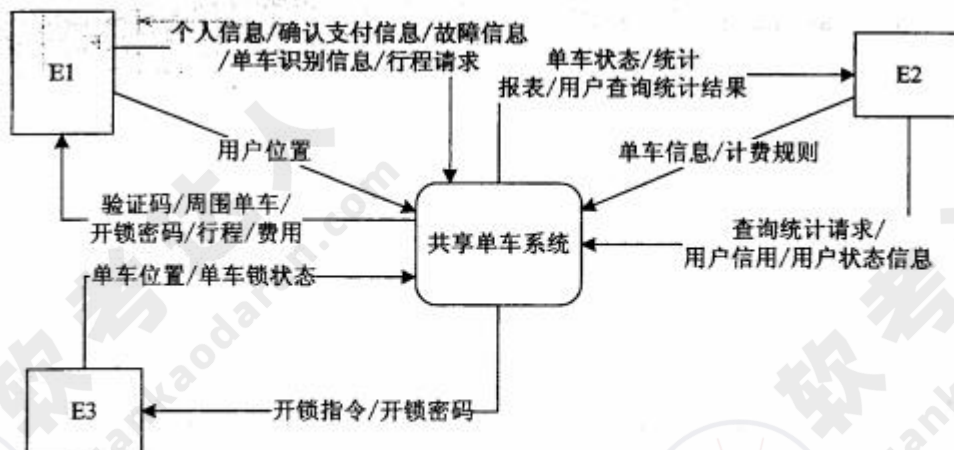
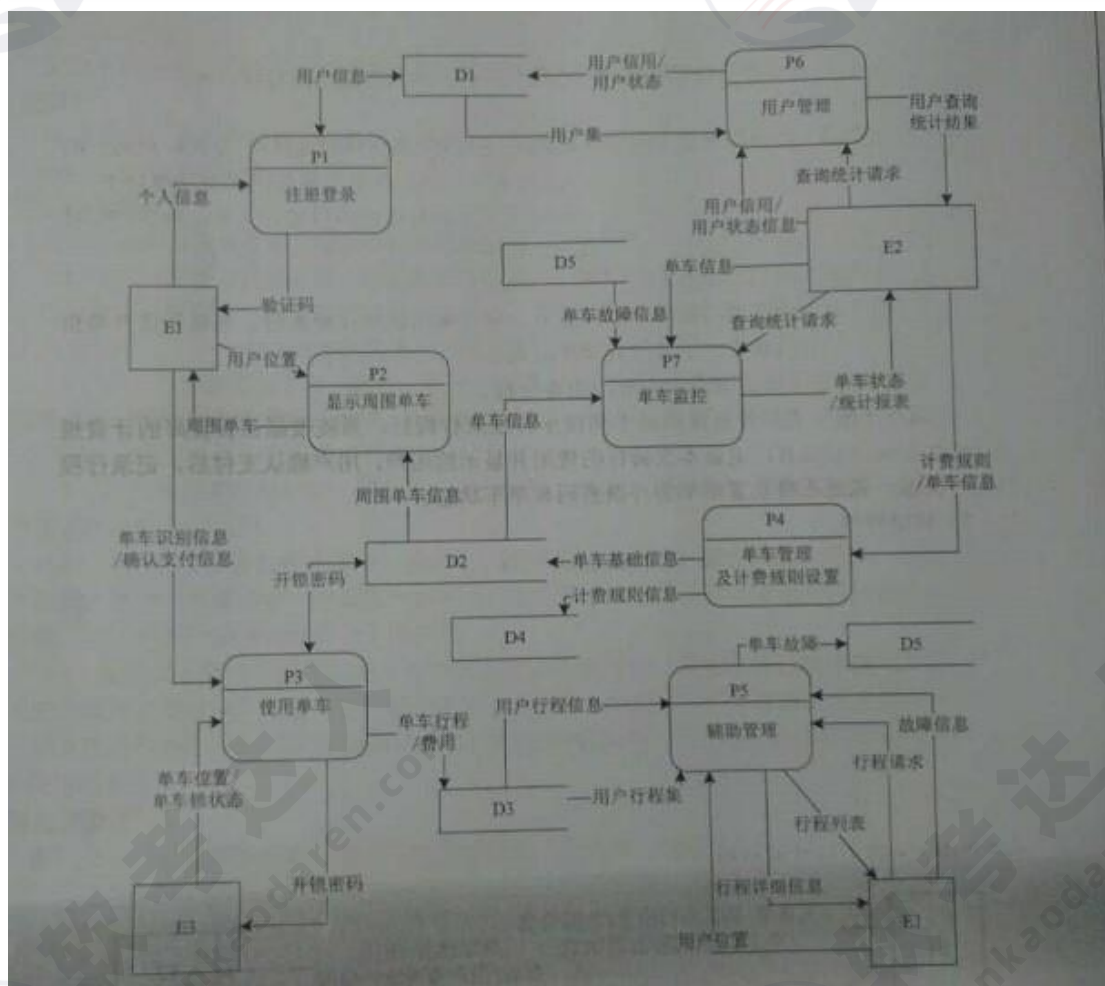


图 1-1 上下文数据流图



【问题 1】(3 分)

使用说明中的词语，给出图 1-1 中的实体 E1~E3 的名称。

E1: 用户

E2: 商家

E3:单车

**【问题 2】**（5 分）

使用说明中的词语，给出图 1-2 中的数据存储 D1~D5 的名称。

D1:存储用户信息

D2:记录单车位置及状态信息

D3:记录单车行程信息

D4:存储计费规则

D5:记录单车故障信息

**【问题 3】**（5 分）

根据说明和图中术语及符号，补充图 1-2 中缺失的数据流及其起点和终点。

1. 起点：p3 终点：E1 数据流名称：开锁密码

2. 起点：p3 终点：E1 数据流名称：锁车结帐

3. 起点：p7 终点：E2 数据流名称：单车状态

**【问题 4】**（2 分）

根据说明中术语，说明“使用单车”可以分解为哪些子加工？

扫码/手动开锁，骑行单车，锁车结帐

## 试题二

阅读下列说明，回答问题 1 至 4 问题，将解答填入答题纸的对应栏内。

### 【说明】

M 公司为了便于开展和管理各项业务活动，提高公司的知名度和影响力，拟构建一个基于网络的会议策划系统。

### 【需求分析结果】

该系统的部分功能及初步需求分析的结果如下：

(1) M 公司旗下有业务部、策划部和其他部门。部门信息包括部门号、部门名、主管、联系电话和邮箱号；每个部门只有一名主管，只负责管理本部门的工作，且主管参照员工关系的员工号；一个部门有多名员工，每名员工属于且仅属于一个部门。

(2) 员工信息包括员工号、姓名、职位、联系方式和薪资。职位包括主管、业务员、策划员等。业务员负责受理用户申请，设置受理标志。一名业务员可以受理多个用户申请，但一个用户申请只能由一名业务员受理。

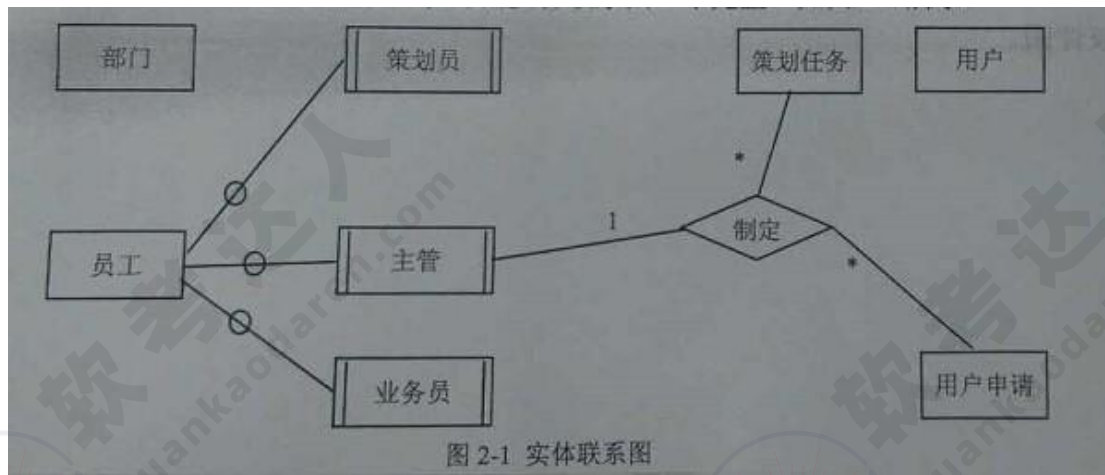
(3) 用户信息包括用户号、用户名、银行账号、电话、联系地址。用户号唯一标识用户信息中的每一个元组。

(4) 用户申请信息包括申请号、用户号、会议日期、天数、参会人数、地点、预算和受理标志。申请号唯一标识用户申请信息中的每一个元组，且一个用户可以提交多个申请，但一个用户申请只对应一个用户号。

(5) 策划部主管为已受理的用户申请制定会议策划任务。策划任务包括申请号、任务明细和要求完成时间。申请号唯一标识策划任务的每一个元组。一个策划任务只对应一个已受理的用户申请，但一个策划任务可由多名策划员参与执行，且一名策划员可以参与执行，且在项策划任务。

### 【概念模型设计】

根据需求阶段收集的信息，设计的实体联系图（不完整）如图 2-1 所示。



### 【关系模型设计】

部门（部门号，部门名，部门主管，联系电话，邮箱号）

员工（员工号，姓名，（ ），联系方式，薪资）

用户（用户名，（ ），电话，联系地址）

用户申请（申请号，用户号，会议日期，天数，参会人数，地点，受理标志，（ ））

执行（申请号，任务明细，（ ））

### 【问题 1】（5 分）

根据问题描述，补充五个联系，完善图 2-1 的实体联系图。联系名可用联系 1、联系 2、联系 3、联系 4 和联系 5，联系的类型为 1:1、1: n 和 m: n（或 1:1、1: \*和\*: \*）。

1. 联系 1：部门和员工，1: n
2. 联系 2：业务员和用户申请，1: n
3. 联系 3：用户和用户申请，1: n
4. 联系 4：策划员和策划任务，n:m
5. 联系 5：策划任务和用户申请，1:1

### 【问题 2】（4 分）

根据题意，将关系模型中的空（a）~（d）补充完整，并填入答题纸对应的位置上。

- a. 职位，部门号
- b. 用户名，银行帐号
- c. 预算费用，业务员(员工号)

d. 要求完成时间

**【问题3】（4分）**

给出“用户申请”和“策划任务”关系模式的主键和外键。

用户申请：主键：申请号 外键：用户号，业务员

策划任务：主键：申请号 外键：申请号

**【问题4】（2分）**

请问“执行”关系模式的主键为全码的说法正确吗？为什么？

不正确。

All-key 关系模型的所有属性组组成该关系模式的候选码，称为全码。即所有属性当作一个码。若关系中只有一个候选码，且这个候选码中包含全部属性，则该候选码为全码。

实际完成时间和用户评价为非主属性。

### 试题三

阅读下列说明，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

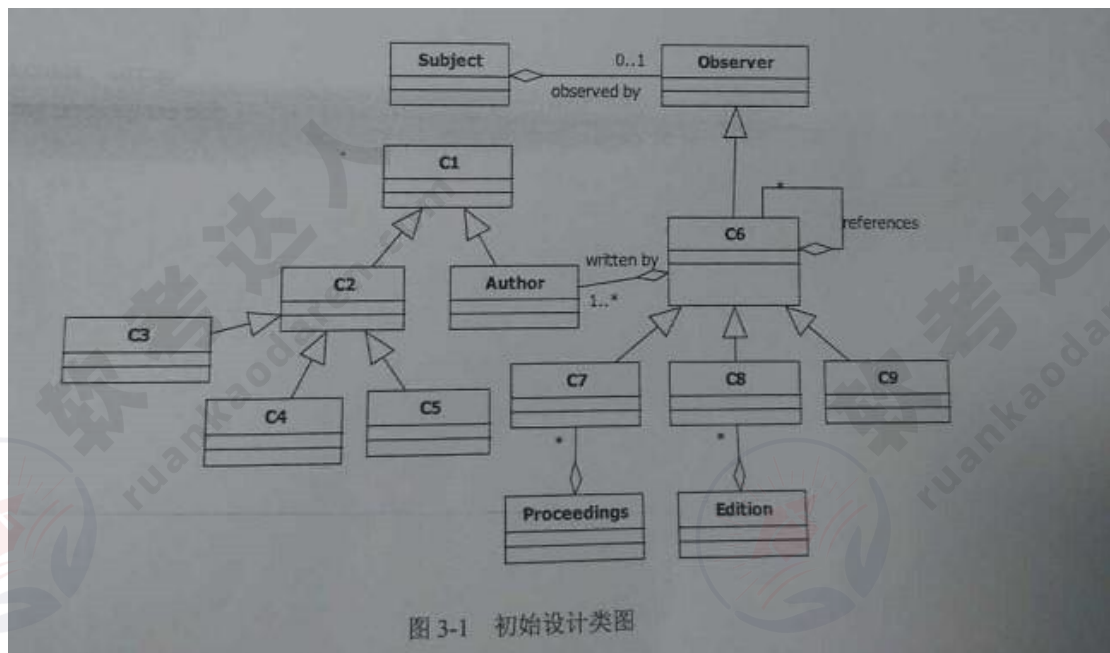
#### 【说明】

某大学拟开发一个用于管理学术出版物（Publication）的数字图书馆系统，用户可以从该系统查询或下载已发表的学术出版物。系统的主要功能如下：

1. 登录系统。系统的用户（User）仅限于该大学的学生（Student）、教师（Faculty）和其他工作人员（Staff）。在访问系统之前，用户必须使用其校园账户和密码登录系统。
2. 查询某位作者（Author）的所有出版物。系统中保存了会议文章（ConfPaper）、期刊文章（JurnalArticle）和校内技术报告（TechReport）等学术出版物的信息，如题目、作者以及出版年份等。除此之外，系统还存储了不同类型出版物的一些特有信息：
  - （1）对于会议文章，系统还记录了会议名称、召开时间以及召开地点；
  - （2）对于期刊文章，系统还记录了期刊名称、出版月份、期号以及主办单位；
  - （3）对于校内技术报告，系统记录了由学校分配的唯一 ID。
3. 查询指定会议集（Proceedings）或某个期刊特定期（Edition）的所有文章。会议集包含了发表在该会议（在某个特定时间段、特定地点召开）上的所有文章。期刊的每一期在特定时间发行，其中包含若干篇文章。
4. 下载出版物。系统记录每个出版物被下载的次数。
5. 查询引用了某篇出版物的所有出版物。在学术出版物中引用他人或早期的文献作为相关工作或背景资料是很常见的现象。用户也可以在系统中为某篇出版物注册引用通知，若有新的出版物引用了该出版物，系统将发送电子邮件通知该用户。

现在采用面向对象方法对该系统进行开发，得到系统的初始设计类图如图 3-1 所示。





【问题 1】(9 分)

根据说明中的描述，给出图 3-1 中 C1~C9 所对应的类名。

C1:Object

C2:User

C3:Student

C4:Faculty

C5:Staff

C6:Publication

C7:ConfPaper

C8:JournalArticle

C9:TechReport

【问题 1】(4 分)

根据说明中的描述，给出图 3-1 中类 C6~C9 的属性。

C6:题目，作者，出版年份

C7:会议名称，召开时间，召开地点

C8:期刊名称，出版月份，期号，主办单位

C9:ID

**【问题3】** （2分）

图 3-1 中包含了哪种设计模式？实现的是该系统的哪个功能？

观察者模式，实现：引用他人学术出版物发送电子邮件通知该用户。

#### 试题四

阅读下列说明和 C 代码，回答问题 1 至问题 2，将解答写在答题纸的对应栏内。

##### 【说明】

一个无向连通图  $G$  点上的哈密尔顿 (Hamilton) 回路是指从图  $G$  上的某个顶点出发，经过图上所有其他顶点一次且仅一次，最后回到该顶点的路劲。一种求解无向图上哈密尔顿回路算法的基础如下如下：

假设图  $G$  存在一个从顶点  $V_0$  出发的哈密尔顿回路  $V_1—V_2—V_3—\dots—V_{n-1}—V_0$ 。

算法从顶点  $V_0$  出发，访问该顶点的一个未被访问的邻接顶点  $V_1$ ，接着从顶点  $V_1$  出发，访问  $V_1$  一个未被访问的邻接顶点  $V_2, \dots$ ；对顶点  $V_i$ ，重复进行以下操作：访问  $V_i$  的一个未被访问的邻接点  $V_{i+1}$ ；若  $V_i$  的所有邻接顶点均已被访问，则返回到顶点  $V_{i-1}$ ，考虑  $V_{i-1}$  的下一个未被访问的邻接顶点，仍记为  $V_i$ ；知道找到一条哈密尔顿回路或者找不到哈密尔顿回路，算法结束。

##### 【C 代码】

下面是算法的 C 语言实现。

(1) 常量和变量说明

$n$  : 图  $G$  中的顶点数

$c[i][j]$  : 图  $G$  的邻接矩阵

$K$ : 统计变量，当期已经访问的定点数为  $k+1$

$x[k]$  : 第  $k$  个访问的顶点编号，从 0 开始

$Visited[x[k]]$ : 第  $k$  个顶点的访问标志，0 表示未访问，1 表示已访问

(2) C 程序

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
#define MAX 100
```

```
Vido Hamilton (int n,int x [MAX],int c[MAX][MAX]) {
```

```
    int t ;
```

```
    int visited[MAX];
```

```
    int k;
```

```
    /*初始化 x 数组和 visited 数组*/
```

```

for (i=0;i<n;i++) {
    x[i]=0;
    visited [i]=0;
}
/*访问起始顶点*/
k=0
( );
x[0]=0
K=k+1
/*访问其他顶点*/
while (k>=0) {
    x[k]=x[k]+1;
    while (x[k]>n) {
        if ( ) &&c[x-[k-1]][x[k]=1) { /*邻接顶点 x[k]未被访问过*/
            break;
        } else {
            x[k] = x[k] +1
        }
    }
    if (x[k] <n-1&& ( ) { /*找到一条哈密尔顿回路*/
        for (k=0;k<n;k++) {
            printf ( "%d-- ", x[k] ; /*输出哈密尔顿回路*/
        }
        printf ( "%d-- ", x[0] ;
        return;
    } else if x[k]<n&&k<n-1) { /*设置当期顶点的访问标志，继续下一个顶点*/
        ( )
        k=k+1;
    } else { /*没有未被访问过的邻接顶点，回退到上一个顶点*/
        x[k]=0;
    }
}

```

```
        visited x[k]=0;
        ( );
    }
}
}
```

**【问题 1】(10 分)**

根据题干说明。填充 C 代码中的空 (1) ~ (5)。

- 1、visited[0] = 1
- 2、visited[x[k]] == 0
- 3、c[x[0]][x[k]]
- 4、visited[x[k]] = 1
- 5、k = k - 1

**【问题 2】(5 分)**

根据题干说明和 C 代码，算法采用的设计策略为 (6)，该方法在遍历图的顶点时，采用的是 (7) 方法（深度优先或广度优先）。

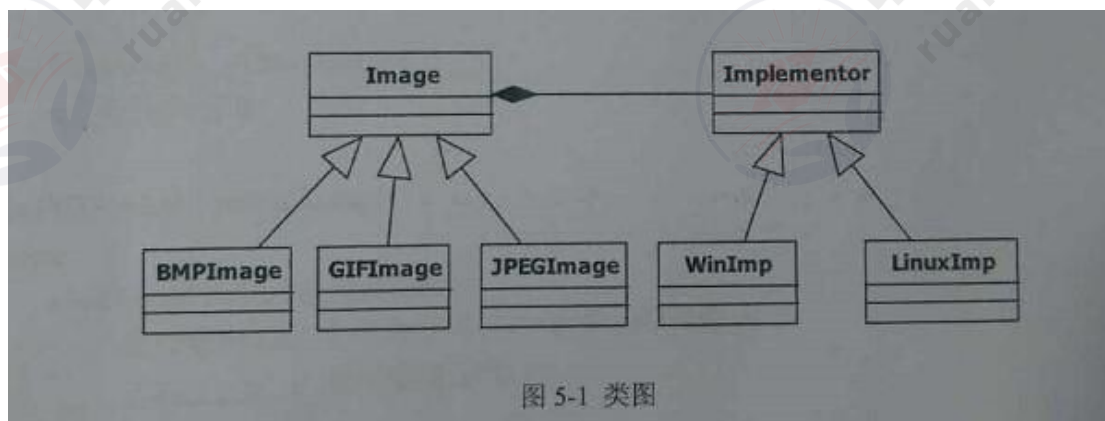
- 6、回溯法
- 7、深度优先

## 试题五

阅读下列说明和 C++代码，将应填入（ ）处的字句写在答题纸的对应栏内。

### 【说明】

某图像预览程序要求能够查看 BMP 、JPEG 和 GIF 三种格式的文件，且能够 Windows 和 Linux 两种操作系统上运行程序需具有较好的扩展性以支持新的文件格式和操作系统。为满足上述需求并减少所需生成的子类数目，现采用桥接（Bridge）模式进行设计，得到如图 5-1 所示的类图。



### 【问题 1】

### 【C++代码】

```

#include

#include<string> ;

Using namespace std;

class Matrix { //各种格式的文件最终都被转化为像素矩阵
    //此处代码省略
};

class Implementor {
public:
    (1) ; 显示像素矩阵 m
};
  
```



```
class WinImp:public Implementor {  
    public:  
        Viod doPaint (Matrix m) {  
            /*调用 Windows 系统的绘制函数绘制像素矩阵*/  
        } ;  
  
class LinuxImp:public Implementor { /*调用 Linux 系统的绘制函数绘制像素矩阵*/  
    } ;  
  
class Image {  
    public:  
        viod setImp (Implement*imp) ? {this->imp=imp;}  
  
        Virtual viod parseFile (string fileName) =0  
  
    protected:  
        Implementor*imp;  
    } ;  
  
class BMPImage:public Image {  
    //此处省略代码  
} ;  
  
class GIFImage:public Image {  
    Public:  
        viod parseFile (string fileName) {  
            //此处解析 GIF 文件并获得一个像素矩阵对象 m  
            (2); 显示像素矩阵 m  
        }  
    } ;  
  
class JPEGImage:public Image {  
    //此处代码省略
```

```
};
```

```
int main () {  
    //在 linux 操作系统上查看 demo.gif 图像文件  
    Image*image= (3) ;  
    Implementor*imageImp= (4) ;  
    (5)  
    Image->parseFile ( "demo.gif " ) ;  
    return 0;  
}
```

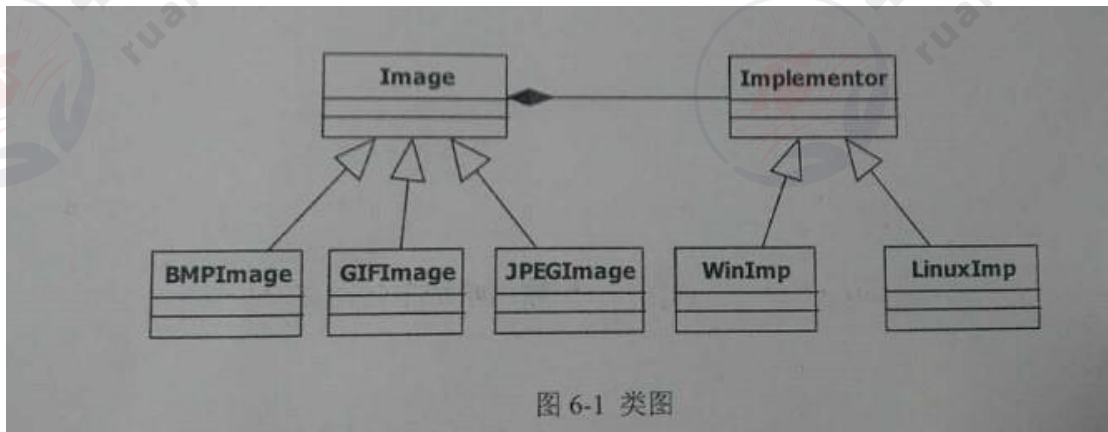
1. virtual void doPaint(Matrix m) = 0
2. imp->doPaint(m)
3. new GIFImage()
4. new LinuxImp()
5. image->setImp(imageImp)

## 试题六

阅读下列说明和 Java 代码，将应填入一旦上一处的字句写在答题纸的对应栏内。

### 【说明】

某图像预览程序要求能够查看 BMP 、JPEG 和 GIF 三种格式的文件，且能够在 Windows 和 Linux 两种操作系统上运行。程序需具有较好的扩展性以支持新的文件格式和操作系统。为满足上述需求并减少所需生成的子类数目，现采用桥接模式进行设计，得到如图 6-1 所示的类图。



### 【问题 1】

#### 【Java 代码】

```

import java.util.*;

class Matrix{ //各种格式的文件最终都被转化为像素矩阵

    //此处代码省略

};

abstract class Implementor {

    Public (1) ;//显示像素矩阵 m

};

class WinImp extends Implementor {

    public void doPaint (Matrix m) { //调用 Windows 系统的绘制函数绘制像素矩阵

    }

};

class LinuxImp extends Implementor{

    public void doPaint (Matrix m) { //调用 Linux 系统的绘制函数绘制像素矩阵
  
```

```

    }
};

abstract class Image {
    public void setImp (Implementor imp) { this.imp= imp; }
    public abstract void parseFile (String fileName) ;
    protected Implementor imp;
};

class BMPImage extends Image {
    //此处代码省略
};

class GIFImage extends Image {
    public void parseFile (String fileName)  {
        //此处解析 BMP 文件并获得一个像素矩阵对象 m
        (2); //显示像素矩阵 m
    }
};

Class Main {
    Public static void main (String[] args) {
        //在 Linux 操作系统上查看 demo.gif 图像文件
        Image image= (3)
        Implementor imageImp= (4)
        (5)
        Image.parseFile ( "demo.gif ");
    }
}

```

1. abstract void doPaint(Matrix m)

2. imp.doPaint(m)

3. new GIFImage()

4. new LinuxImp()

5. image.setImp(imageImp)