

对有 n 个结点、 e 条边且采用数组表示法（即邻接矩阵存储）的无向图进行深度优先遍历, 时间复杂度为 (1)。

- (1) A. $O(n^2)$ B. $O(e^2)$ C. $O(n+e)$ D. $O(n*e)$

【答案】A

【解析】

在邻接表中, 就是要依次访问每个顶点, 然后在每个顶点中依次访问每条边, 把这些边的终点的入度+1。也就是每个顶点和每条边依次要各访问一遍, 所以时间复杂度是 $O(n+e)$ 。在邻接矩阵中, 算法需要遍历邻接矩阵的每一个点, 而邻接矩阵有 $n*n$ 个点, 所以时间复杂度是 $O(n*n)$ 。

浮点数的表示分为阶和尾数两部分。两个浮点数相加时, 需要先对阶, 即 (2) (n 为阶差的绝对值)。

- (2) A. 将大阶向小阶对齐, 同时将尾数左移 n 位
B. 将大阶向小阶对齐, 同时将尾数右移 n 位
C. 将小阶向大阶对齐, 同时将尾数左移 n 位
D. 将小阶向大阶对齐, 同时将尾数右移 n 位

【答案】D

【解析】

对阶时, 小数向大数看齐; 对阶是通过较小数的尾数右移实现的。

计算机运行过程中, 遇到突发事件, 要求 CPU 暂时停止正在运行的程序, 转去为突发事件服务, 服务完毕, 再自动返回原程序继续执行, 这个过程称为 (3), 其处理过程中保存现场的目的是 (4)。

- (3) A. 阻塞 B. 中断 C. 动态绑定 D. 静态绑定
(4) A. 防止丢失数据 B. 防止对其他部件造成影响
C. 返回去继续执行原程序 D. 为中断处理程序提供数据

【答案】B C

【解析】

中断是指计算机运行过程中, 出现某些意外情况需主机干预时, 机器能自动停止正在运行的程序并转入处理新情况的程序, 处理完毕后又返回原被暂停的程序继续运行。

海明码是一种纠错码，其方法是为需要校验的数据位增加若干校验位，使得校验位的值决定于某些被校位的数据，当被校数据出错时，可根据校验位的值的变化找到出错位，从而纠正错误。对于 32 位的数据，至少需要加 (5) 个校验位才能构成海明码。

以 10 位数据为例，其海明码表示为 D9D8D7D6D5D4P4D3D2D1P3D0P2P1 中，其中 D_i ($0 \leq i \leq 9$) 表示数据位， P_j ($1 \leq j \leq 4$) 表示校验位，数据位 D9 由 P4、P3 和 P2 进行校验（从右至左 D9 的位序为 14，即等于 $8+4+2$ ，因此用第 8 位的 P4、第 4 位的 P3 和第 2 位的 P2 校验），数据位 D5 由 (6) 进行校验

- (5) A. 3 B. 4 C. 5 D. 6
- (6) A. P4P1 B. P4P2 C. P4P3P1 D. P3P2P1

【答案】D B

【解析】本问考查的是海明编码的规则

(5) 海明不等式：校验码个数为 k ，2 的 k 次方个校验信息，1 个校验信息用来指出“没有错误”，满足 $m+k+1 \leq 2^k$ 。所以 32 位的数据位，需要 6 位校验码。

(6) 构造监督关系式，和校验码的位置相关：数据位 D9 受到 P4、P3、P2 监督 ($14=8+4+2$)，那么 D5 受到 P4、P2 的监督 ($10=8+2$)。

流水线的吞吐率是指单位时间流水线处理的任务数，如果各段流水的操作时间不同，则流水线的吞吐率是 (7) 的倒数。

- (7) A. 最短流水段操作时间 B. 各段流水的操作时间总和
- C. 最长流水段操作时间 D. 流水段数乘以最长流水段操作时间

【答案】C

【解析】

流水线处理机在执行指令时，把执行过程分为若干个流水级，若各流水级需要的时间不同，则流水线必须选择各级中时间较大者为流水级的处理时间。

理想情况下，当流水线充满时，每一个流水级时间流水线输出一个结果。

流水线的吞吐率是指单位时间流水线处理机输出的结果的数目，因此流水线的吞吐率为一个流水级时间的倒数，即最长流水级时间的倒数。

网络管理员通过命令行方式对路由器进行管理，要确保 ID，口令和会话话内存的保密

性，应采取的访问方式是(8)。

- (8) A. 控制台 B. AUX C. TELNET D. SSH

【答案】D

【解析】

SSH 为 Secure Shell 的缩写，由 IETF 的网络小组（Network Working Group）所制定；SSH 为建立在应用层基础上的安全协议。SSH 是目前较可靠，专为远程登录会话和其他网络服务提供安全性的协议。利用 SSH 协议可以有效防止远程管理过程中的信息泄露问题。

在安全通信中，S 将所发送的信息使用(9)进行数字签名，T 收到该消息后可利用(10)验证该消息的真实性。

- (9) A. S 的公钥 B. S 的私钥 C. T 的公钥 D. T 的私钥
(10) A. S 的公钥 B. S 的私钥 C. T 的公钥 D. T 的私钥

【答案】B A

【解析】

数字签名技术是将摘要信息用发送者的私钥加密，与原文一起传送给接收者。接收者只有用发送者的公钥才能解密被加密的摘要信息，然后用 HASH 函数对收到的原文产生一个摘要信息，与解密的摘要信息对比。如果相同，则说明收到的信息是完整的，在传输过程中没有被修改，否则说明信息被修改过，因此数字签名能够验证信息的完整性。

数字签名是个加密的过程，数字签名验证是个解密的过程。保证信息传输的完整性、发送者的身份认证、防止交易中的抵赖发生。

在网络安全管理中，加强内防内控可采取的策略有(11)。

- ①控制终端接入数量
- ②终端访问授权，防止合法终端越权访问
- ③加强终端的安全检查与策略管理
- ④加强员工上网行为管理与违规审计

- (11) A. ②③ B. ②④ C. ①②③④ D. ②③④

【答案】D

【解析】

内务内控管理，主要是为了管理内部网络，防止越权访问，以及内部泄露信息。

攻击者通过发送一个目的主机已经接收过的报文来达到攻击目的，这种攻击方式属于(12)攻击。

- (12) A. 重放 B. 拒绝服务 C. 数据截获 D. 数据流分析

【答案】A

【解析】

重放攻击（Replay Attacks）又称重播攻击、回放攻击，是指攻击者发送一个目的主机已接收过的包，来达到欺骗系统的目的，主要用于身份认证过程，破坏认证的正确性。重放攻击可以由发起者，也可以由拦截并重发该数据的敌方进行。

以下有关计算机软件著作权的叙述中，正确的是(13)。

- (13) A. 非法进行拷贝、发布或更改软件的人被称为软件盗版者
B. 《计算机软件保护条例》是国家知识产权局颁布的, 用来保护软件著作权人的权益
C. 软件著作权属于软件开发者，软件著作权自软件开发完成之日起产生
D. 用户购买了具有版权的软件，则具有对该软件的使用权和复制权

【答案】A

【解析】

选项 B 中由国务院颁布；

选项 C 中委托开发、合作开发软件著作权的归属及行使原则与一般作品著作权归属及行使原则一样，但职务计算机软件的著作权归属有一定的特殊性。自然人在法人或者其他组织中任职期间所开发的软件有下列情形之一的，该软件著作权由该法人或者其他组织享有，该法人或者其他组织可以对开发软件的自然人进行奖励。

选项 D 中复制权，是指制作作品复制品的权利，依作品表现形式不同分为三种情形：（1）以图书、报纸、期刊等印刷品形式复制和传播作品的权利，即通常所说的出版权；（2）以唱片、磁带、幻灯片等音像制品形式复制和传播作品的权利，即录音录像权或机械复制权；（3）使用临摹、照相、雕塑、雕刻等方法复制和传播美术等作品的权利，即狭义上的复制权。而购买软件版权不享有复制权。

王某是某公司的软件设计师，完成某项软件开发后按公司规定进行软件归档。以下有关

该软件的著作权的叙述中，正确的是(14)。

- (14) A. 著作权应由公司和王某共同享有
B. 著作权应由公司享有
C. 著作权应由王某享有
D. 除署名权以外，著作权的其它权利由王某享有

【答案】B

【解析】

此为职务作品，凡是供职于某公司，利用公司相关资源进行开发完成的作品，其著作权归公司所有。

著作权中，(15)的保护期不受限制。

- (15) A. 发表权 B. 发行权 C. 署名权 D. 展览权

【答案】C

【解析】

我国《著作权法》对著作权的保护期限作了如下规定：著作权中的署名权、修改权、保护作品完整权的保护期不受限制。

数据字典是结构化分析的一个重要输出。数据字典的条目不包括(16)。

- (16) A. 外部实体 B. 数据流 C. 数据项 D. 基本加工

【答案】A

【解析】

数据字典是指对数据的数据项、数据结构、数据流、数据存储、处理逻辑、外部实体等进行定义和描述，其目的是对数据流程图中的各个元素做出详细的说明，使用数据字典为简单的建模项目。其条目有数据流、数据项、数据存储、基本加工等。

某商店业务处理系统中，基本加工“检查订货单”的描述为：若订货单金额大于 5000 元，且欠款时间超过 60 天，则不予批准；若订货单金额大于 5000 元，且欠款时间不超过 60 天，则发出批准书和发货单；若订货单金额小于或等于 500 元，则发出批准书和发货单，若欠款时间超过 60 天，则还要发催款通知书。现采用决策表表示该基本加工，则条件取值的组合数最少是(17)。

(17) A. 2

B. 3

C. 4

D. 5

【答案】 B

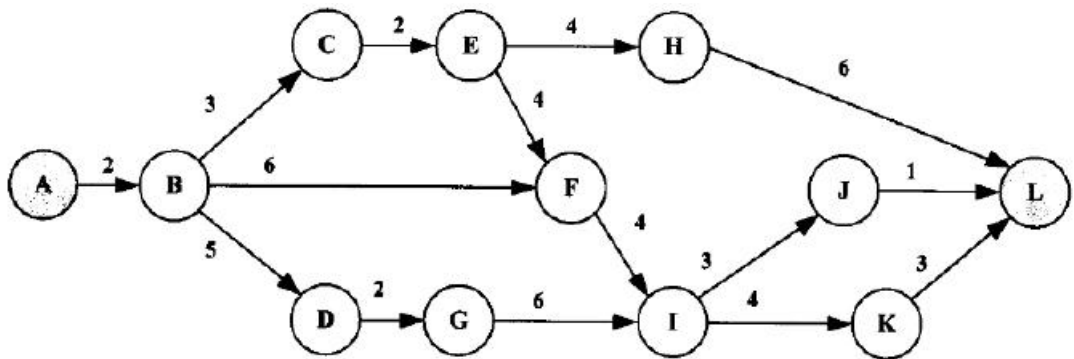
【解析】

根据题意可得出如下决策表

条件项	条件值			
	1	2	3	4
>5000 元	Y	Y	N	N
>60 天	Y	N	Y	N
批准书	N	Y	Y	Y
发货单	N	Y	Y	Y
催款通知书	N	N	Y	N

其中第 2 条和第 4 条可进行合并，故该条件取值的组数为 3。

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, 连接顶点的边表示包含的活动, 边上的数字表示活动的持续天数, 则完成该项目的最少时间为 (18) 天。活动 EH 和 IJ 的松弛时间分别为 (19) 天。



(18) A. 17

B. 19

C. 20

D. 22

(19) A. 3 和 3

B. 3 和 6

C. 5 和 3

D. 5 和 6

【答案】 D C

【解析】

在网络图中的某些活动可以并行地进行，所以完成工程的最少时间是从开始顶点到结束顶点的最长路径长度，从开始顶点到结束顶点的最长（工作时间之和最大）路径为关键路径，关键路径上的活动为关键活动。

本题关键路径为：A-B-D-G-I-K-L，共 22 天。

工作量估算模型 COCOMO II 的层次结构中,估算选择不包括(20)。

(20) A. 对象点 B. 功能点 C. 用例数 D. 源代码行

【答案】C

【解析】

COCOMOII 模型也需要使用规模估算信息，在模型层次结构中有 3 种不同规模估算选择，即：对象点、功能点和代码行。

(21) 是一种函数式编程语言。

(21) A. Lisp B. Prolog C. Python D. Java/C++

【答案】A

【解析】

LISP 是一种通用高级计算机程序语言，长期以来垄断人工智能领域的应用。LISP 作为因应人工智能而设计的语言，是第一个声明式系内函数式程序设计语言，有别于命令式系内过程式的 C、Fortran 和面向对象的 Java、C#等结构化程序设计语言。

将高级语言源程序翻译为可在计算机上执行的形式有多种不同的方式，其中 (22)。

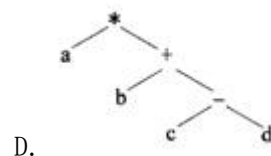
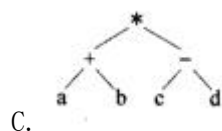
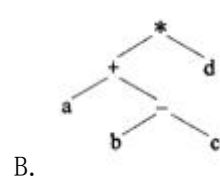
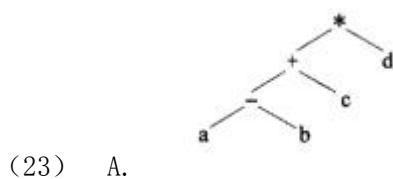
- (22) A. 编译方式和解释方式都生成逻辑上与源程序等价的目标程序
B. 编译方式和解释方式都不生成逻辑上与源程序等价的目标程序
C. 编译方式生成逻辑上与源程序等价的目标程序, 解释方式不生成
D. 解释方式生成逻辑上与源程序等价的目标程序, 编译方式不生成

【答案】C

【解析】

编译语言是一种以编译器来实现的编程语言。它不像直译语言一样，由解释器将代码一句一句运行，而是以编译器，先将代码编译为机器码，再加以运行。将某一种程序设计语言写的程序翻译成等价的另一种语言的程序的程序，称之为编译程序。

对于后缀表达式 $a \ b \ c \ - \ + \ d \ *$ (其中, -, +, *表示二元算术运算减、加、乘), 与该后缀式等价的语法树为 (23)。

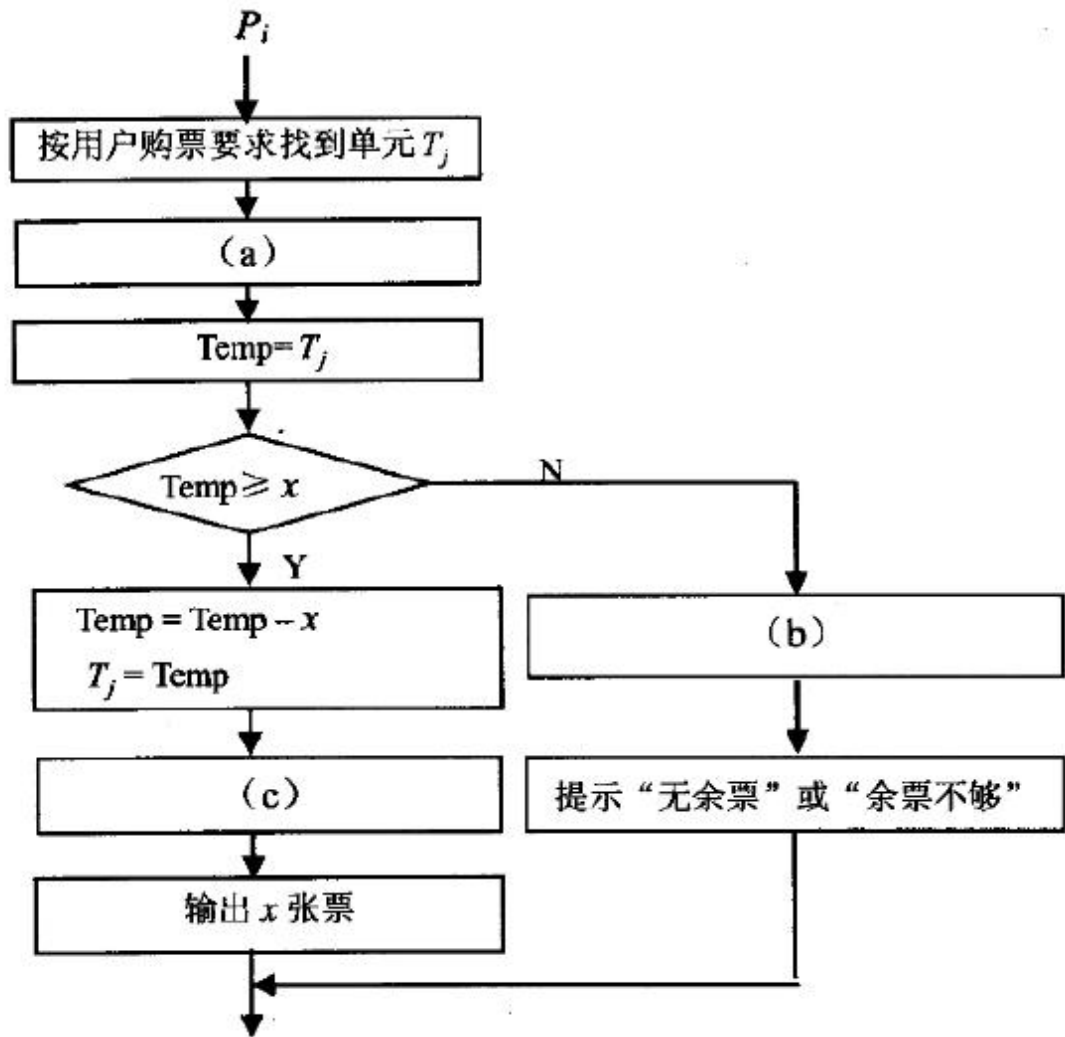


【答案】B

【解析】

对题中选项中 4 个二叉树进行后序遍历,得出结果与该后缀表达式一致的则为与其等价的语法树。

假设铁路自动售票系统有 n 个售票终端,该系统为每个售票终端创建一个进程 P_i ($i=1, 2, \dots, n$) 管理车票销售过程。假设 T_j ($j=1, 2, \dots, m$) 单元存放某日某趟车的车票剩余票数, $Temp$ 为 P_i 进程的临时工作单元, x 为某用户的购票张数。 P_i 进程的工作流程如下图所示,用 P 操作和 V 操作实现进程间的同步与互斥。初始化时系统应将信号量 S 赋值为(24)。图中 (a)、(b) 和 (c) 处应分别填入(25)。



- (24) A. $n-1$ B. 0 C. 1 D. 2
- (25) A. V(S)、P(S) 和 P(S) B. P(S)、P(S) 和 V(S)
- C. V(S)、V(S) 和 P(S) D. P(S)、V(S) 和 V(S)

【答案】C D

【解析】

(24) 信号量 S 应当是该单元数，对某日某趟车为一个单元的话，单元数只能为 1。

(25) (a) 应为申请资源，(b) (c) 应当为释放资源，故是一个 P，两个 V 操作。

若系统在将 (26) 文件修改的结果写回磁盘时发生崩溃，则对系统的影响相对较大。

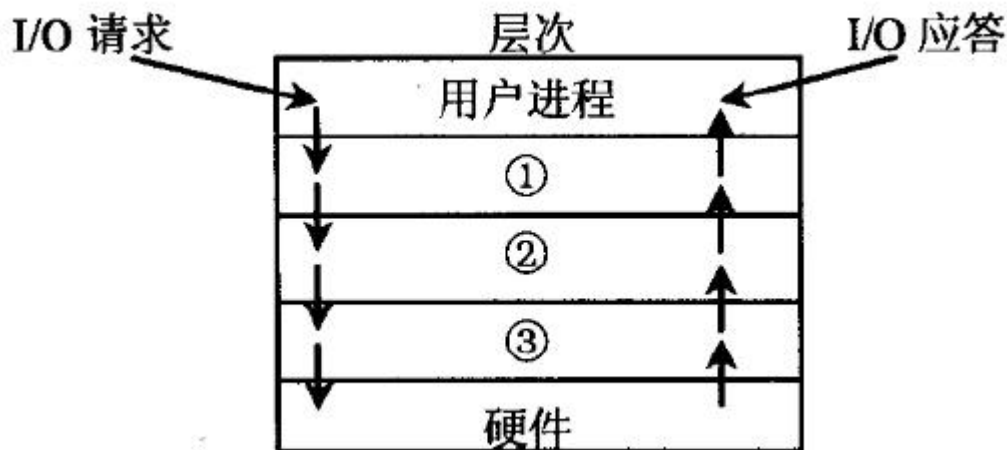
- (26) A. 目录 B. 空闲块 C. 用户程序 D. 用户数据

【答案】A

【解析】

系统目录就是指操作系统的主要文件存放的目录，目录中的文件直接影响到系统是否正常工作。

I/O 设备管理软件一般分为 4 个层次，如下图所示。图中①②③分别对应 (27)。



- (27) A. 设备驱动程序、虚设备管理、与设备无关的系统软件
B. 设备驱动程序、与设备无关的系统软件、虚设备管理
C. 与设备无关的系统软件、中断处理程序、设备驱动程序
D. 与设备无关的系统软件、设备驱动程序、中断处理程序

【答案】D

【解析】

具体层次从上往下分别为用户级 I/O 层、设备无关 I/O 层、设备驱动程序、中断处理程序、硬件。

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。

若某文件系统的目录结构如下图所示，假设用户要访问文件 `rw.d11`，且当前工作目录为 `stools`，则该文件的全文件名为 (28)，相对路径和绝对路径分别为 (29)。

瀑布模型的流程实施项目开发，以方便对项目的管理；但在软件的实际创建中，则将软件系统按功能分解为许多增量构件，并以构件为单位逐个地创建与交付，直到全部增量构件创建完毕，并都被集成到系统之中交付用户使用。比较瀑布模型、原型进化模型，增量模型具有非常显著的优越性。但增量模型对软件设计有更高的技术要求，特别是对软件体系结构，要求它具有很好的开放性与稳定性，能够顺利地实现构件的集成。

本文档由微信号:ruankaopass，一手整理，通过他人购买的，拒绝售后。本人专业提供软考历年真题

能力成熟度模型集成（CMMI）是若干过程模型的综合和改进。连续式模型和阶段式模型是 CMMI 提供的两种表示方法。连续式模型包括 6 个过程域能力等级（Capability Level, CL），其中 (31) 的共性目标是过程将可标识的输入工作产品转换成可标识的输出工作产品，以实现支持过程域的特定目标。

(31) A. CL1（已执行的）

B. CL2（已管理的）

C. CL3（已定义的）

D. CL4（定量管理的）

【答案】A

【解析】

本题考查能力成熟度模型。

软件维护工具不包括 (32) 工具。

(32) A. 版本控制

B. 配置管理

C. 文档分析

D. 逆向工程

【答案】B

【解析】

辅助软件维护过程中的活动的软件称为“软件维护工具”，它辅助维护人员对软件代码及其文档进行各种维护活动。软件维护工具主要有：1、版本控制工具；2、文档分析工具；3、开发信息库工具；4、逆向工程工具；5、再工程工具；6、配置管理支持工具。

概要设计文档的内容不包括 (33)。

(33) A. 体系结构设计

B. 数据库设计

C. 模块内算法设计

D. 逻辑数据结构设计

【答案】C

【解析】

一般来讲，概要设计的内容可以包含系统构架、模块划分、系统接口、数据设计 4 个主

要方面的内容，不包括模块内算法设计。

耦合是模块之间的相对独立性（互相连接的紧密程度）的度量。耦合程度不取决(34)。

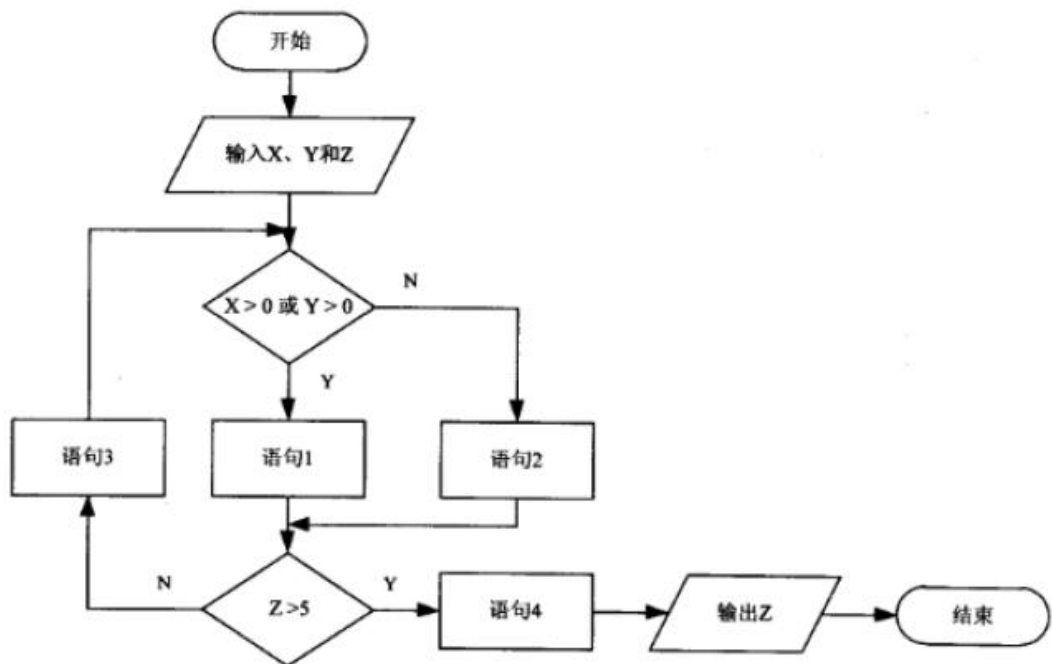
- (34) A. 调用模块的方式
B. 各个模块之间接口的复杂程度
C. 通过接口的信息类型
D. 模块提供的功能数

【答案】D

【解析】

耦合性也叫块间联系。指软件系统结构中各模块间相互联系紧密程度的一种度量。模块之间联系越紧密，其耦合性就越强，模块之间越独立则越差，模块间耦合的高低取决于模块间接口的复杂性，调用的方式以及传递的信息。

对下图所示的程序流程图进行判定覆盖测试,则至少需要(35)个测试用例。采用 McCabe 度量法计算其环路复杂度为 (36)。



- (35) A. 2 B. 3 C. 4 D. 5
- (36) A. 2 B. 3 C. 4 D. 5

【答案】 A B

【解析】

(35)判定覆盖是设计足够多的测试用例,使得程序中的每一个判断至少获得一次“真”

和一次“假”，即使得程序流程图中的每一个真假分支至少被执行一次。根据题意，只需 2 个测试用例即可。

(36) 根据环路复杂度的计算公式 $V(G) = m - n + 2 = 11 - 10 + 2 = 3$ 。

软件调试的任务就是根据测试时所发现的错误,找出原因和具体的位置,进行改正。其常用的方法中,(37)是指从测试所暴露的问题出发,收集所有正确或不正确的数据,分析它们之间的关系,提出假想的错误原因,用这些数据来证明或反驳,从而查出错误所在。

(37) A. 试探法 B. 回溯法 C. 归纳法 D. 演绎法

【答案】C

【解析】

所谓演绎推理,就是从一般性的前提出发,通过推导即“演绎”,得出具体陈述或个别结论的过程。归纳法以一系列经验事物或知识素材为依据,寻找出其服从的基本规律或共同规律,并假设同类事物中的其他事物也服从这些规律,从而将这些规律作为预测同类事物的其他事物的基本原理的一种认知方法。

对象的(38)标识了该对象的所有属性(通常是静态的)以及每个属性的当前值(通常是动态的)。

(38) A. 状态 B. 唯一 ID C. 行为 D. 语义

【答案】A

【解析】

对象的状态包括这个对象的所有属性(通常是静态的)以及每个属性当前的值(通常是动态的);为了将一个对象与其它所有对象区分开来,我们通常会给它起一个“标识”;行为是对象根据它的状态改变和消息传递所采取的行动和所作出的反应;对象的行为代表了其外部可见的活动;操作代表了一个类提供给它的对象的一种服务。

在下列机制中,(39)是指过程调用和响应调用所需执行的代码在运行时加以结合;而(40)是过程调用和响应调用所需执行的代码在编译时加以结合。

(39) A. 消息传递 B. 类型检查 C. 静态绑定 D. 动态绑定

(40) A. 消息传递 B. 类型检查 C. 静态绑定 D. 动态绑定

【答案】D C

【解析】

程序运行过程中,把函数(或过程)调用与响应调用所需要的代码相结合的过程称为动态绑定。静态绑定是指在程序编译过程中,把函数(方法或者过程)调用与响应调用所需的代码相结合的过程称之为静态绑定。

同一消息可以调用多种不同类的对象的方法,这些类有某个相同的超类,这种现象是(41)。

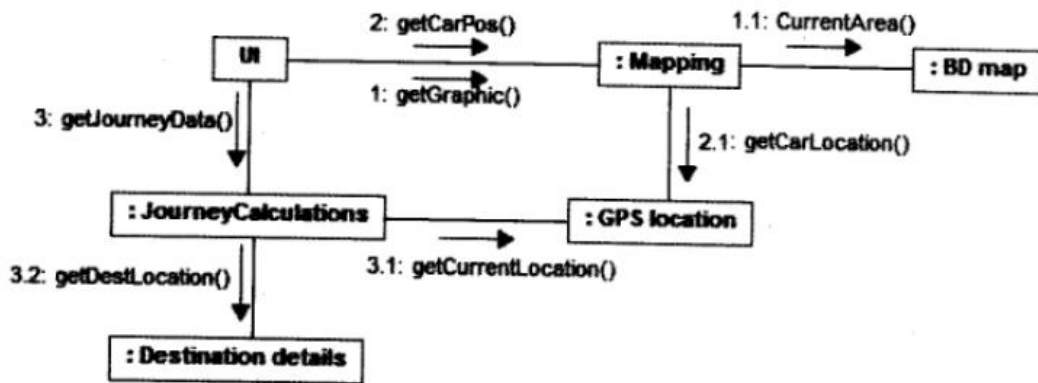
- (41) A. 类型转换 B. 映射 C. 单态 D. 多态

【答案】D

【解析】

多态指相同的对象收到不同的消息或者不同的对象收到相同的消息时产生的不同的实现动作。

如下所示的图为UML的(42),用于展示某汽车导航系统中(43)。 Mapping对象获取汽车当前位置(GPS Location)的消息为(44)。



- (42) A. 类图 B. 组件图 C. 通信图 D. 部署图

- (43) A. 对象之间的消息流及其顺序 B. 完成任务所进行的活动流
C. 对象的状态转换及其事件顺序 D. 对象之间消息的时间顺序

- (44) A 1: getGraphic () B. 2: getCarPos ()
C. 1.1: CurrentArea () D 2.1: getCarLocation ()

【答案】C A D

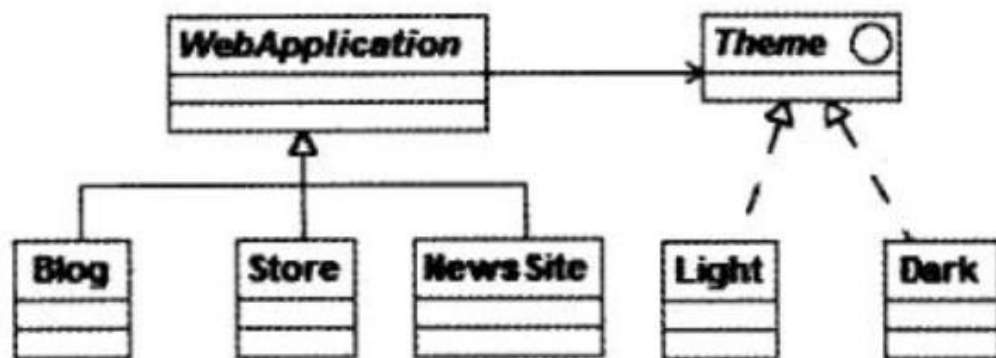
【解析】

(42、43) 通信图 (communication diagram) 是一种交互图,它强调收发消息的对象

或参与者的结构组织。顺序图和通信图表达了类似的基本概念，但它们所强调的概念不同，顺序图强调的是时序，通信图强调的是对象之间的组织结构（关系）。

（44）获取汽车当前位置的消息为 2.1: getCarLocation()。

假设现在要创建一个Web应用框架, 基于此框架能够创建不同的具体Web应用, 比如博客、新闻网站和网上商店等; 并可以为每个Web应用创建不同的主题样式, 如浅色或深色等。这一业务需求的类图设计适合采用 (45) 模式 (如下图所示)。其中 (46) 是客户程序使用的主要接口, 维护对主题类型的引用。此模式为 (47), 体现的最主要的意图是 (48)。



(45) A. 观察者(Observer) B. 访问者(Visitor) C. 策略(Strategy) D. 桥接(Bridge)

(46) A. Webapplication B. Blog C. Theme D. Light

(47) 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

(48) A. 将抽象部分与其实实现部分分离, 使它们都可以独立地变化

B. 动态地给一个对象添加一些额外的职责

C. 为其他对象提供一种代理以控制对这个对象的访问

D. 将一个类的接口转换成客户希望的另外一个接口

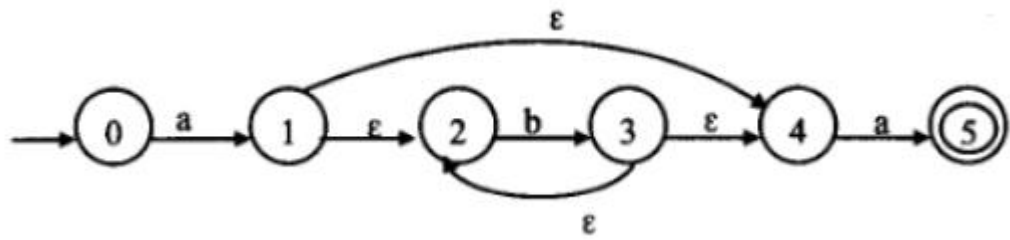
【答案】D A B A

【解析】

桥接模式是一种结构型设计模式。Bridge 模式基于类的最小设计原则, 通过使用封装、聚合及继承等行为让不同的类承担不同的职责。将类的抽象部分和它的实现部分分离开来, 使它们可以独立地变化。

下图所示为一个不确定有限自动机 (NFA) 的状态转换图。该 NFA 识别的字符串集合可

用正规式(49)描述。



- (49) A. ab^*a B. $(ab)^*a$ C. a^*ba D. $a(ba)^*$

【答案】A

【解析】

根据图中展示，其正规式的收尾必定会包含 a 和 b 字符，答案是 ab^*a 。

简单算术表达式的结构可以用下面的上下文无关文法进行描述（E 为开始符号），(50)是符合该文法的句子。

$E \rightarrow T \mid E+T$

$T \rightarrow F \mid T * F$

$F \rightarrow -F \mid N$

$N \rightarrow 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$

- (50) A. $2--3*4$ B. $2+-3*4$ C. $(2+3)^*4$ D. $2*4-3$

【答案】B

【解析】

一个上下文无关语法定义一个语言，其主要思想是从文法的开始符号出发，反复连续使用产生式，对非终结符进行替换和展开。

语法制导翻译是一种(51)方法。

- (51) A. 动态语义分析 B. 中间代码优化 C. 静态语义分析 D. 目标代码优化

【答案】C

【解析】

翻译的任务：首先是语义分析和正确性检查，若正确，则翻译成中间代码或目标代码。其基本思想是，根据翻译的需要设置文法符号的属性，以描述语法结构的语义。例如，一个变量的属性有类型，层次，存储地址等。表达式的属性有类型，值等。属性值的计算和产生式相联系。随着语法分析的进行，执行属性值的计算，完成语义分析和翻译的任务。

给定关系模式 $R\langle U, F \rangle$, 其中 U 为属性集, F 是 U 上的一组函数依赖, 那么 Armstrong 公理系统的伪传递律是指 (52)。

- (52) A. 若 $X \rightarrow Y, X \rightarrow Z$, 则 $X \rightarrow YZ$ 为 F 所蕴涵
B. 若 $X \rightarrow Y, WY \rightarrow Z$, 则 $XW \rightarrow Z$ 为 F 所蕴涵
C. 若 $X \rightarrow Y, Y \rightarrow Z$ 为 F 所蕴涵, 则 $X \rightarrow Z$ 为 F 所蕴涵
D. 若 $X \rightarrow Y$ 为 F 所蕴涵, 且 $Z \subseteq U$, 则 $XZ \rightarrow YZ$ 为 F 所蕴涵

【答案】B

【解析】

从已知的一些函数依赖, 可以推导出另外一些函数依赖, 这就需要一系列推理规则。函数依赖的推理规则最早出现在 1974 年 W.W.Armstrong 的论文里, 这些规则常被称作“Armstrong 公理”。

设 U 是关系模式 R 的属性集, F 是 R 上成立的只涉及 U 中属性的函数依赖集。函数依赖的推理规则有以下三条:

自反律: 若属性集 Y 包含于属性集 X , 属性集 X 包含于 U , 则 $X \rightarrow Y$ 在 R 上成立。(此处 $X \rightarrow Y$ 是平凡函数依赖)

增广律: 若 $X \rightarrow Y$ 在 R 上成立, 且属性集 Z 包含于属性集 U , 则 $XZ \rightarrow YZ$ 在 R 上成立。

传递律: 若 $X \rightarrow Y$ 和 $Y \rightarrow Z$ 在 R 上成立, 则 $X \rightarrow Z$ 在 R 上成立。

根据上面三条推理规则, 又可推出下面三条推理规则:

合并规则: 若 $X \rightarrow Y, X \rightarrow Z$, 则 $X \rightarrow YZ$ 为 F 所蕴含;

伪传递规则: 若 $X \rightarrow Y, WY \rightarrow Z$, 则 $XW \rightarrow Z$ 为 F 所蕴含;

分解规则: 若 $X \rightarrow Y, Z \subseteq Y$, 则 $X \rightarrow Z$ 为 F 所蕴含。

给定关系 $R(A, B, C, D, E)$ 与 $S(B, C, F, G)$, 那么与表达式 $\pi_{2, 4, 6, 7}(\sigma_{2 < 7}(R \Join S))$ 等价的 SQL 语句如下:

SELECT (53) FROM R, S WHERE (54);

- (53) A. R.B, D, F, G
B. R.B, E, S.C, F, G
C. R.B, R.D, S.C, F
D. R.B, R.C, S.C, F

- (54) A. R.B=S.B OR R.C=S.C OR R.B < S.G
B. R.B=S.B OR R.C=S.C OR R.B < S.C

C. R.B=S.B AND R.C=S.C AND R.B <S.G

D. R.B=S.B AND R.C=S.C AND R.B <S.C

【答案】C C

【解析】

自然连接是一种特殊的等值连接,它要求两个关系中进行比较的分量必须是相同的属性组,并且在结果中把重复的属性列去掉。经过自然连接后的结果属性有(A, S. B, S. C, D, E, F, G);投影运算就是从表中选择需要的属性列,第 2, 4, 6, 7 列分别为(R. B, D, F, G);选择运算 $\sigma_{2<7}$ (R??S) 的意思在 R??S 的结果中,选择出满足属性列 2 的值 <属性列 7 的值的行。

给定教师关系 Teacher (T_no, T_name, Dept_name, Tel), 其中属性 T_no、 T_name、 Dept_name 和 Tel 的含义分别为教师号、教师姓名、学院名和电话号码。用 SQL 创建一个“给定学院名求该学院的教师数”的函数如下:

```
Create function Dept_count (Dept_name varchar(20))
```

(55)

```
begin
```

(56)

```
select count(*) into d_count
```

```
from Teacher
```

```
where Teacher.Dept_ name= Dept_name
```

```
return d_count
```

```
end
```

(55) A. returns integer

B. returns d_count integer

C. declare integer

D. declare d_count integer

(56) A. returns integer

B. returns d_count integer

C. declare integer

D. declare d_count integer

【答案】A D

【解析】

(55) 第 1 空最后应当加一个 as, 也可缺省。为函数的返回值类型, 即为 integer。

(56) 第 2 空为声明 d_count。

微信号: ruankaopass, 淘宝店铺: 软考真题教育, 专业提供软考历年真题

某集团公司下属有多个超市, 每个超市的所有销售数据最终要存入公司的数据仓库中。假设该公司高管需要从时间、地区和商品种类三个维度来分析某家电商品的销售数据, 那么最适合采用 (57) 来完成。

- (57) A. Data Extraction B. OLAP C. OLTP D. ETL

【答案】B

【解析】

联机分析处理 OLAP 是一种软件技术, 它使分析人员能够迅速、一致、交互地从各个方面观察信息, 以达到深入理解数据的目的。

队列的特点是先进先出, 若用循环单链表表示队列, 则 (58)。

- (58) A. 入队列和出队列操作都不需要遍历链表
B. 入队列和出队列操作都需要遍历链表
C. 入队列操作需要遍历链表而出队列操作不需要
D. 入队列操作不需要遍历链表而出队列操作需要

【答案】B

【解析】

循环单链表中最后一个结点的指针域不再是结束标志, 而是指向整个链表的第一个结点, 从而使链表形成一个环。

设有 n 阶三对角矩阵 A , 即非零元素都位于主对角线以及与主对角线平行且紧邻的两条对角线上, 现对该矩阵进行按行压缩存储, 若其压缩空间用数组 B 表示, A 的元素下标从 0 开始, B 的元素下标从 1 开始。已知 $A[0, 0]$ 存储在 $B[1]$, $A[n-1, n-1]$ 存储在 $B[3n-2]$, 那么非零元素 $A[i, j]$ ($0 \leq i < n, 0 \leq j < n, |i-j| \leq 1$) 存储在 $B[(59)]$ 。

- (59) A. $2i+j-1$ B. $2i+j$ C. $2i+j+1$ D. $3i-j+1$

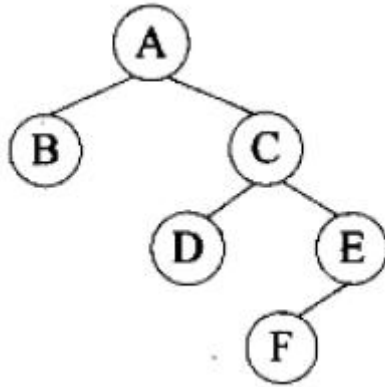
【答案】C

【解析】

将 $i=0, j=0$ 和 $i=n-1, j=n-1$ 分别代入 4 个选项中, 使其分别满足结果 1 和 $3n-2$ 的为正确答案。

对下面的二叉树进行顺序存储 (用数组 MEM 表示), 已知结点 A、B、C 在 MEM 中对应元

素的下标分别为 1、2、3, 那么结点 D、E、F 对应的数组元素下标为 (60)。



(60) A. 4、5、6

B. 4、7、10

C. 6、7、8

D. 6、7、14

【答案】D

【解析】

二叉树的顺序存储,就是用一组连续的存储单元存放二叉树中的结点;把二叉树的所有结点安排成为一个恰当的序列,反映出节点中的逻辑关系;用编号的方法从树根起,自上层至下层,每层自左至右地给所有结点编号。

用哈希表存储元素时,需要进行冲突(碰撞)处理,冲突是指 (61)。

(61) A. 关键字被依次映射到地址编号连续的存储位置

B. 关键字不同的元素被映射到相同的存储位置

C. 关键字相同的元素被映射到不同的存储位置

D. 关键字被映射到哈希表之外的位置

【答案】B

【解析】

哈希法又称散列法、杂凑法以及关键字地址计算法等,相应的表成为哈希表。其基本思想:首先在元素的关键字 K 和元素的位置 P 之间建立一个对应关系 f ,使得 $P=f(K)$,其中 f 成为哈希函数。创建哈希表时,把关键字 K 的元素直接存入地址为 $f(K)$ 的单元;查找关键字 K 的元素时利用哈希函数计算出该元素的存储位置 $P=f(K)$;当关键字集合很大时,关键字值不同的元素可能会映像到哈希表的同一地址上,即 $K_1 \neq K_2$,但 $f(K_1)=f(K_2)$,这种现象称为 hash 冲突,实际中冲突是不可避免的,只能通过改进哈希函数的性能来减少冲突。

现需要申请一些场地举办一批活动,每个活动有开始时间和结束时间。在同一个场地,

如果一个活动结束之前,另一个活动开始,即两个活动冲突。若活动 A 从 1 时间开始,5 时间结束,活动 B 从 5 时间开始,8 时间结束,则活动 A 和 B 不冲突。现要计算 n 个活动需要的最少场地数。

求解该问题的基本思路如下(假设需要场地数为 m ,活动数为 n ,场地集合为 P_1, P_2, \dots, P_m),初始条件 P_i 均无活动安排:

(1) 采用快速排序算法对 n 个活动的开始时间从小到大排序,得到活动 a_1, a_2, \dots, a_n 。
对每个活动 a_i , i 从 1 到 n ,重复步骤 (2)、(3) 和 (4);

(2) 从 p_1 开始,判断 a_i 与 P_1 的最后一个活动是否冲突,若冲突,考虑下一个场地 P_2, \dots ;

(3) 一旦发现 a_i 与某个 P_j 的最后一个活动不冲突,则将 a_i 安排到 P_j ,考虑下一个活动;

(4) 若 a_i 与所有已安排活动的 P_j 的最后一个活动均冲突,则将 a_i 安排到一个新的场地,考虑下一个活动;

(5) 将 n 减去没有安排活动的场地数即可得到所用的最少场地数

算法首先采用了快速排序算法进行排序,其算法设计策略是 (62);后面步骤采用的算法设计策略是 (63)。整个算法的时间复杂度是 (64)。下表给出了 $n=11$ 的活动集合,根据上述算法,得到最少的场地数为 (65)。

i	1	2	3	4	5	6	7	8	9	10	11
开始时间 s_i	0	1	2	3	3	5	5	6	8	8	12
结束时间 f_i	6	4	13	5	8	7	9	10	11	12	14

(62) A. 分治 B. 动态规划 C. 贪心 D. 回溯

(63) A. 分治 B. 动态规划 C. 贪心 D. 回溯

(64) A. $\Theta(\lg n)$ B. $\Theta(n)$ C. $\Theta(n \lg n)$ D. $\Theta(n^2)$

(65) A. 4 B. 5 C. 6 D. 7

【答案】A C C B

【解析】

(62) 快速排序采用的是分治法,其基本思想是将原问题分解成若干个规模更小但结构与原问题相似的子问题。

(63) 通过递归地解决这些子问题,然后再将这些子问题的解组合成原问题的解。

(64) 其时间复杂度为 $\Theta(n \log^2 n)$ 。

(65) 贪心法总是做出在当前来说是最好的选择, 而并不从整体上加以考虑, 它所做的每步选择只是当前步骤的局部最优选择, 但从整体来说不一定是最优的选择。根据其算法可得, P1 场地可排 a1、a8、a11; P2 场地可排 a2、a6、a9; P3 场地可排 a3; P4 场地可排 a4、a7; P5 场地可排 a5、a10; 最少需要的场地数为 5。

下列网络互连设备中, 属于物理层的是 (66)。

- (66) A. 交换机 B. 中继器 C. 路由器 D. 网桥

【答案】B

【解析】

中继器 (RP repeater) 是工作在物理层上的连接设备。适用于完全相同的两类网络的互连, 主要功能是通过将数据信号的重新发送或者转发, 来扩大网络传输的距离。 中继器是对信号进行再生和还原的网络设备: OSI 模型的物理层设备。

在地址 `http:// wwwdailynews.com.cn/channel/welcome.htm` 中, `wwwdailynews.com.cn` 表示 (67), `welcome.htm` 表示 (68)。

- (67) A. 协议类型 B. 主机 C. 网页文件名 D. 路径

- (68) A. 协议类型 B. 主机域名 C. 网页文件名 D. 路径

【答案】B C

【解析】

(67) 域名地址用来在 Internet 上唯一确定位置的地址, 通常用来指明所使用的计算机资源位置及查询信息的类型。其由字母或数字组成, 中间以 “.” 隔开, `www.dailynews.com.cn` 表示访问的主机和域名。

(68) `welcome.htm` 表示网页文件名。

在 Linux 中, 要更改一个文件的权限设置可使用 (69) 命令。

- (69) A. `attrib` B. `modify` C. `chmod` D. `change`

【答案】C

【解析】

修改 Linux 文件权限命令: `chmod`。

主域名服务器在接收到域名请求后, 首先查询的是 (70)。

(70) A. 本地 hosts 文件 B. 转发域名服务器 C. 本地缓存 D. 授权域名服务器

【答案】C

【解析】

域名解析流程:

1. 客户机提出域名解析请求, 并将该请求发送给本地的域名服务器。
2. 当本地的域名服务器收到请求后, 就先查询本地的缓存, 如果有该纪录项, 则本地的域名服务器就直接把查询的结果返回。
3. 如果本地的缓存中没有该纪录, 则本地域名服务器就直接把请求发给根域名服务器, 然后根域名服务器再返回给本地域名服务器一个所查询域(根的子域)的主域名服务器的地址。
4. 本地服务器再向上一步返回的域名服务器发送请求, 然后接受请求的服务器查询自己的缓存, 如果没有该纪录, 则返回相关的下级的域名服务器的地址。
5. 重复第 4 步, 直到找到正确的纪录。
6. 本地域名服务器把返回的结果保存到缓存, 以备下一次使用, 同时还将结果返回给客户机。

Creating a clear map of where the project is going is an important first step. It lets you identify risks, clarify objectives, and determine if the project even makes sense. The only thing more important than the release plan is not to take it too seriously.

Release planning is creating a game plan for your Web project (71) what you think you want your Web site to be. The plan is a guide for the content, design elements, and functionality of a Web site to be released to the public, to partners, or internally. It also (72) how long the project will take and how much it will cost. What the plan is not is a functional (73) that defines the project in detail or that produces a budget you can take to the bank.

Basically you use a release Plan to do an initial sanity check of the project's (74) and worthiness. Release Plans are useful road maps, but don't think of them as guides to the interstate road system. Instead, think of them as the (75) used

by early explorers--half umor and guess and half hope and expectation.

It's always a good idea to have a map of where a project is headed

(71) A. constructing B. designing C. implementing D. outlining

(72) A. defines B. calculates C. estimates D. knows

(73) A. specification B. structure

C. requirement D. implementation

(74) A. correctness B. modifiability C. feasibilit D. traceability

(75) A. navigators B. maps C. guidant D. goals

【答案】 D C A C B

【解析】

试题一（共 15 分）

阅读下列说明，回答间顺顺，将解答填入题纸的对应栏内。

【说明】

某医疗护理机构为老年人或有护理需求者提供专业护理，现欲开发一基于 web 的医疗管理系统，以改善医疗护理效，该系统的主要要功能如下：

- （1）通用信息查询。客广提交通用信息查询请求，意而通用信表表，返回查询结果
- （2）医生聘用。医生提出成 / 职中申，交由去前行用事事，更新医生表，并给医生反聘用解聘结果：聊除解聘医生的出诊发排
- （3）预约处理。医生安排出诊时向间，存入医生出诊时间表，根据客户提交的预约查询请求，查在职医生及其出诊时间等约所需数据并返同问：创建预约，提交预约请求，在在预约表中新增预约记录，更新所约医生出诊时间并给医生发送约通知给客户反馈预约结果。
- （4）药品管理。医生提交处方，根据药品名称从药品数据中查询相关药品库存信息开出药品，更新对应药品的库存以及预约表中的治疗信息；给医生发送“药品已开出”反馈。
- （5）报表创建。根据主管提交的报表查询请求（报表类型和时间段），从预的数、通用信息、药品库存数据、医生以及医生出诊时间中进行查询，生成报表返回给主管。

现采用结构化方法对医疗管理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。

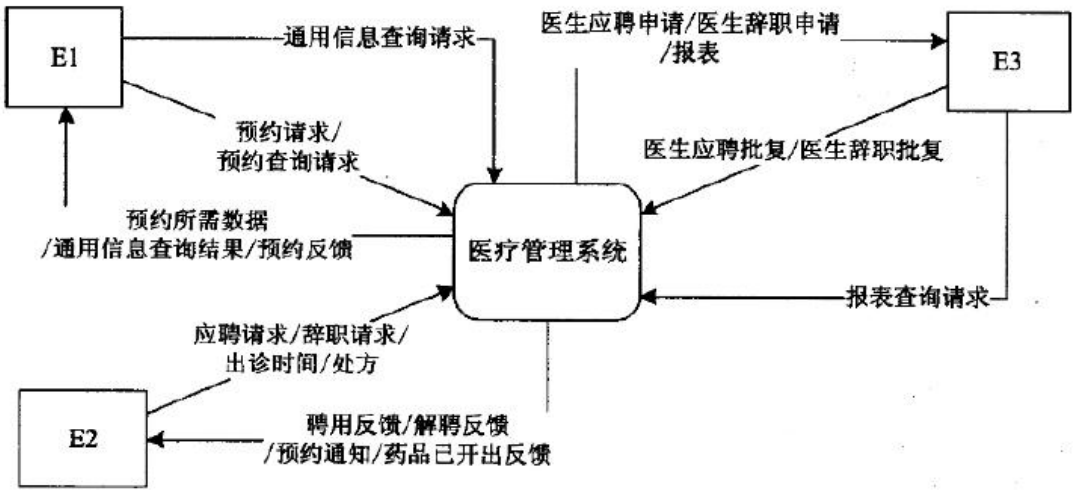


图 1-1 上下文数据流图

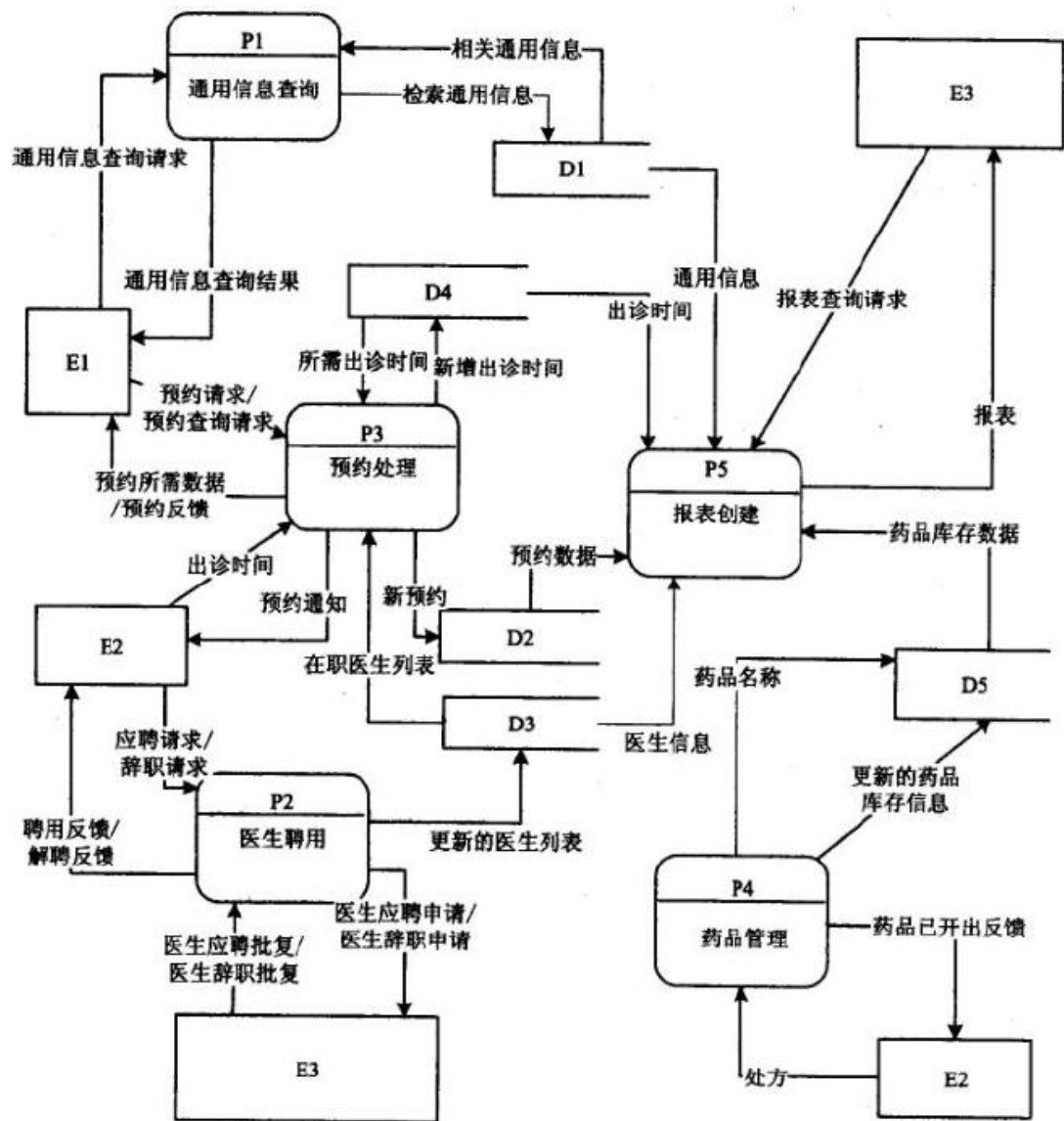


图 1-20 层数据流图

【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。

【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。

【问题 3】(4 分)

使用说明和图中术语，补充图 1-2 中缺失的数据流及其起点和终点。

【问题 4】(3 分)

使用说明中的词语，说明“预约处理”可以分解为哪些子加工，并说明建模图 1-1 和图 1-2 是如何保持数据流图平衡。

【答案】

问题 1:

E1: 客户 E2: 医生 E3: 主管

问题 2:

D1: 通用信息表

D2: 预约表

D3: 医生表

D4: 出诊时间表

D5: 药品库存表

问题 3:

数据流名称	起点	终点
删除解聘医生的出诊安排	P2	D4
查询相关药品库存信息	P4	D2
更新预约表中治疗信息	D5	P4
更新所约医生出诊时间	P3	D4

问题 4:

预约处理分解为: 安排出诊、创建预约、提交预约请求、发送预约通知。

即保持父图与子图之间的平衡: 父图中某个加工的输入输出数据流必须与其子图的输入输出数据流在数量上和名字上相同。父图的一个输入(或输出)数据流对应于子图中几个输入(或输出)数据流, 而子图中组成的这些数据流的数据项全体正好是父图中的这一个数据流。

试题二（共 15 分）

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】

某海外代购公司为扩展公司业务，需要开发一个信息化管理系统。请根据公司现有业务及需求完成该系统的数据库设计。

【需求描述】

- （1）记录公司员工信息。员工信息包括工号、身份证号、姓名、性别和一个手机号，工号唯一标识每位员工，员工分为代购员和配送员。
- （2）记录采购的商品信息。商品信息包括商品名称、所在超市名称、采购价格、销售价格和商品介绍，系统内部用商品条码唯一标识每种商品。一种商品只在一家超市代购。
- （3）记录顾客信息。顾客信息包括顾客真实姓名、身份证号（清关缴税用）、一个手机号和一个收货地址，系统自动生成唯一的顾客编号。
- （4）记录托运公司信息。托运公司信息包括托运公司名称、电话和地址，系统自动生成唯一的托运公司编号。
- （5）顾客登录系统之后，可以下订单购买商品。订单支付成功后，系统记录唯一的支付凭证编号，顾客需要在订单里指定运送方式：空运或海运。
- （6）代购员根据顾客的订单在超市采购对应商品，一份订单所含的多个商品可能由多名代购员从不同超市采购。
- （7）采购完的商品交由配送员根据顾客订单组合装箱，然后交给托运公司运送。托运公司按顾客订单核对商品名称和数量，然后按顾客的地址进行运送。

【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。

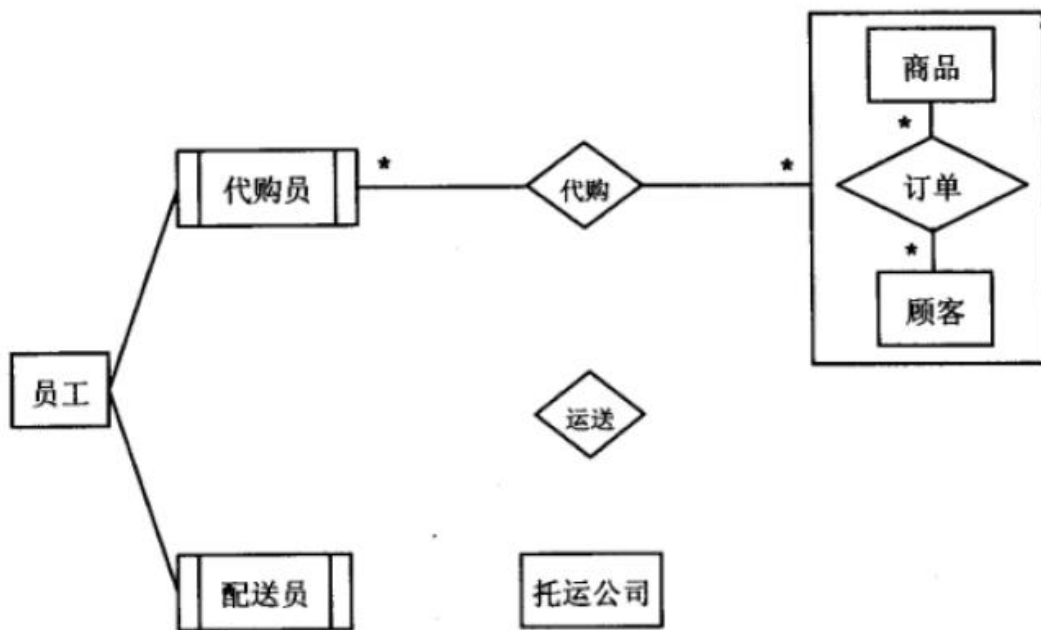


图 2-1 实体联系图

【逻辑结构设计】

据概念模型设计阶段完成的实体联系图，得出如下关系模式（不完整）：

员工（工号，身份证号，姓名，性别，手机号）

商品（条码，商品名称，所在超市名称，采购价格，销售价格，商品介绍）

顾客（编号，姓名，身份证号，手机号，收货地址）

托运公司司（托运公司编号，托运公司名称，电话，地址）

订单（订单 ID，(a)，商品数量，运送方式，支付凭证编号）

代购（代购 ID，代购员工号，(b)）

运送（运送 D，配送员工号，托运公司编，订单 D，发运时间）

【问题 1】（3 分）

根据问题描述，补充图 2—1 的实体联系图。

【问题 2】（6 分）

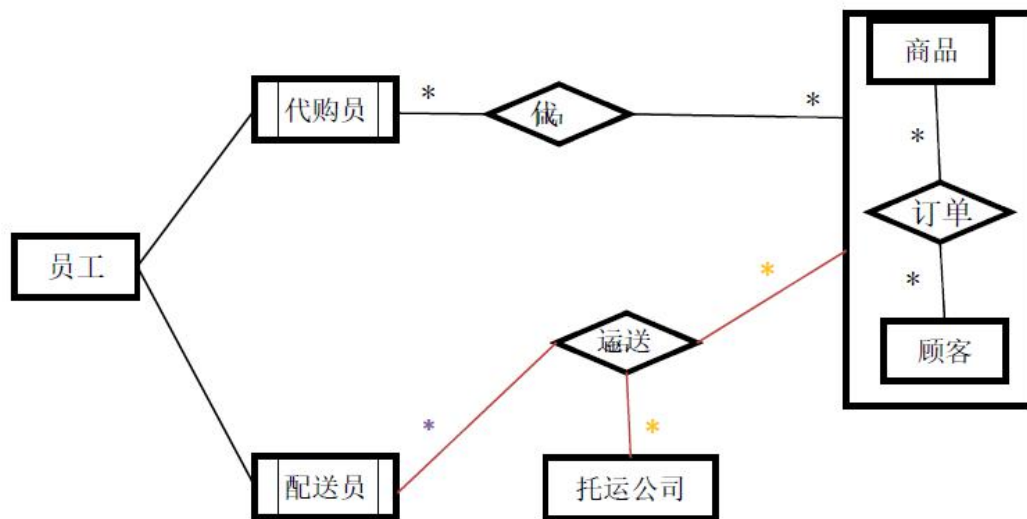
补充逻辑结构设计结果中的 (a)、(b) 两处空缺。

【问题 3】（6 分）

为方便顾客，允许顾客在系统中保存多组收货地址。请根据此需求，增加“顾客地址”弱实体，对图 2—1 进行补充，并修改“运送”关系模式。

【答案】

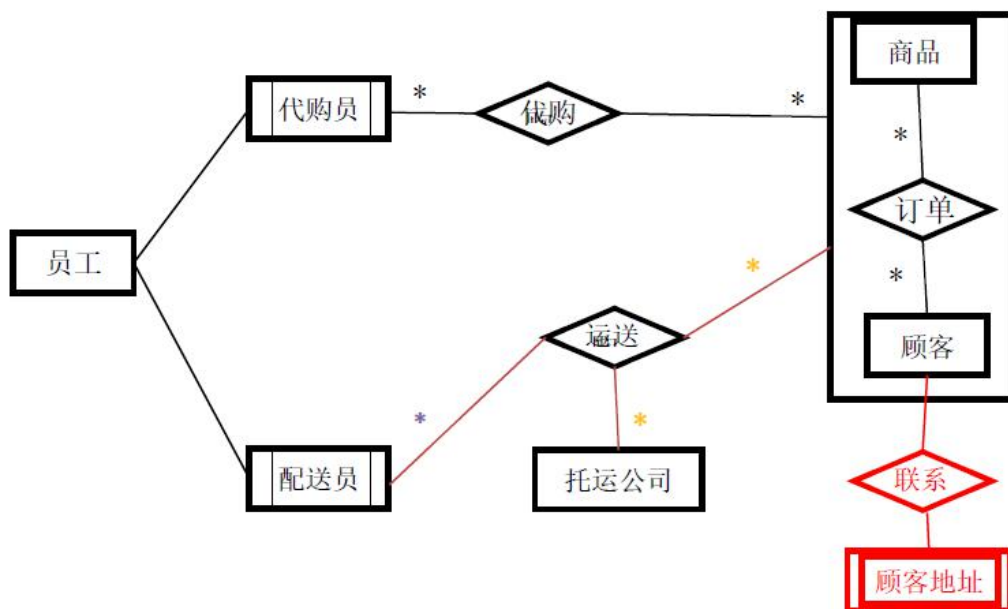
问题 1:



问题 2:

- (a) 商品条码, 顾客编号
- (b) 订单 ID, 商品条码

问题 3:



新增一个弱实体顾客地址, 新增一个联系 客户收货地址, 联连接顾客实体和顾客地址

类型为 1: *; 弱实体用双矩型

运送关系模式增加该订单收货地址

试题三（共 15 分）

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 ETC（Electronic Toll Collection，不停车收费）系统在高速公路沿线的特定位置上设置一个横跨道路上空的龙门架（Toll gantry），龙门架下包括 6 条车道（Traffic lanes），每条车道上安装有雷达传感器（Radar sensor）、无线传输器（Radio transceiver）和数码相机（Digital Camera）等用于不停车收费的设备，以完成正常行驶速度下的收费工作。该系统的基本工作过程如下：

（1）每辆汽车上安装有车载器，驾驶员（Driver）将一张具有唯一识别码的磁卡插入车载器中。磁卡中还包含有驾驶员账户的当前信用记录。

（2）当汽车通过某条车道时，不停车收费设备识别车载器内的特有编码，判断车型，将收集到的相关信息发送到该路段所属的区域系统（Regional center）中，计算通行费用创建收费交易（Transaction），从驾驶员的专用账户中扣除通行费用。如果驾驶员账户透支，则记录透支账户交易信息。区域系统再将交易后的账户信息发送到维护驾驶员账户信息的中心系统（Central system）

（3）车载器中的磁卡可以使用邮局的付款机进行充值。充值信息会传送至中心系统，以更新驾驶员账户的余额。

（4）当没有安装车载器或者车载器发生故障的车辆通过车道时，车道上的数码相机将对车辆进行拍照，并将车辆照片及拍摄时间发送到区域系统，记录失败的交易信息；并将该交易信息发送到中心系统。

（5）区域系统会获取不停车收费设备所记录的交通事件（Traffic events）；交通广播电台（Traffic advice center）根据这些交通事件进行路况分析并播报路况。现采用面向对象方法对上述系统进行分析与设计，得到如表 3—1 所示的用例列表以及如图 3—1 所示的用例图和图 3—2 所示的分析类图。

表 3-1 用例列表

用例名称	说明
Create transaction	记录收费交易
Charge card	磁卡充值
Underpaid transaction	记录透支账户交易信息
Record Illegal use	记录失败交易信息
Record traffic event	记录交通事件

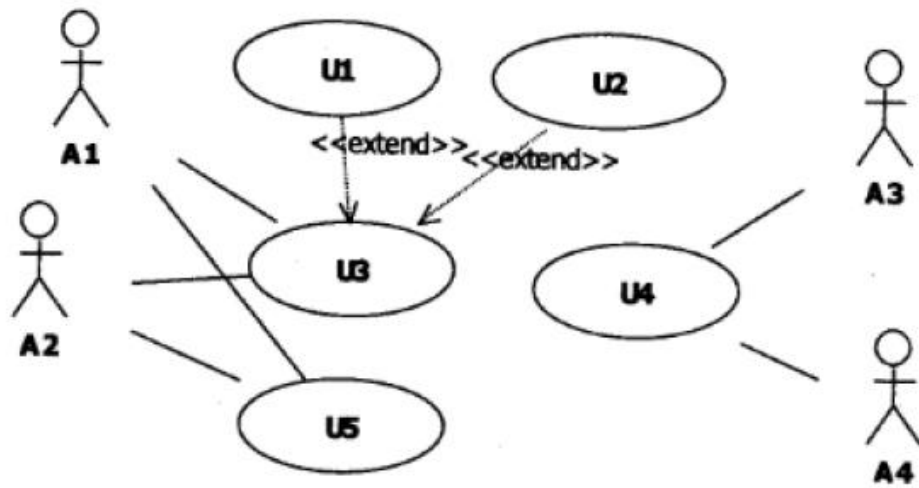


图 3-1 用例图

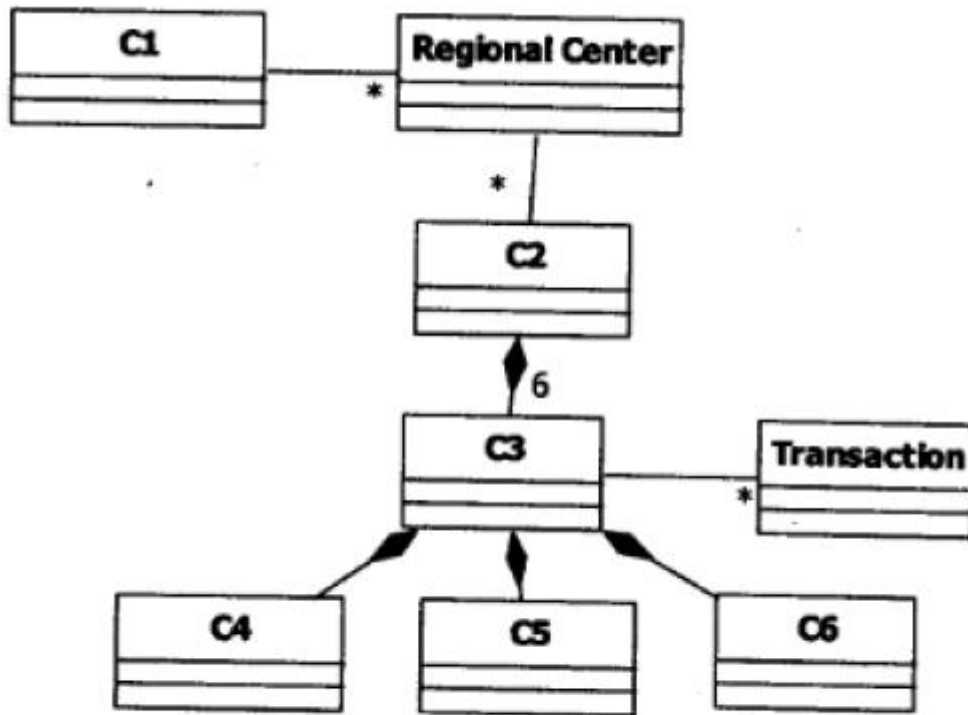


图 3-2 分析类图

【问题 1】(4 分)

根据说明中的描述，给出图 3-1 中 A1~A4 所对应的参与者名称。

【问题 2】(5 分)

根据说明中的描述及表 3-1，给出图 3-1 中 UI-U5 所对应的用例名称。

【问题 3】(6 分)

根据说明中的描述，给出图 3-2 中 C1~C6 所对应的类名。

【答案】

问题 1:

A1: Central system 或中心系统

A2: Driver 或驾驶员

A3: Regional center 或区域系统

A4: Traffic advice center 或交通广播电台

其中 A3、A4 可以互换

本文档由微信号:ruankaopass, 一手整理, 通过他人购买的, 拒绝售后。本人专业提供软考历年真题

问题 2:

U1: Underpaid transaction

U2: Record Illegal use

U3: Create transaction

U4: Record traffic event

U5: Charge card

其中 U1、U2 可以互换, 用例名称必须为英文, 因为表中的汉字是对用例的说明。

问题 3:

C1: Center system

C2: Traffic events

C3: Toll gantry

C4: Radar sensor

C5: Radio transceiver

C6: Digital Camera

其中 C4、C5、C6 可以互换

【解析】

问题 1: 首先要识别出参与者, 是人或组织。题干中表示参与者的有: 驾驶员、区域系统、中心系统、交通广播电台。由问题 2 得出 A1 为中心系统。

问题 2: 填写用例, 要先找到突破口, 本题有两个突破口, 一个是 U1、U2 是 U3 的扩展, 而所给定的用例名称中, 只有记录透支帐户交易信息和记录失败交易信息, 是记录收费交易的扩展。二个是有一个孤立的用例, 从题干中可知为记录交通事件, 再加上问题 1 中的参与

者，可以推出完整的用例图。

问题 3：先从 C4、C5、C6 与 C3 的一个多组合关系。题干中只有龙门架由三个部分组成。C1 与 Regional Center 对应关系是 1 个对象对应多个对象，C1 只可能为中心系统。然后题干(5)中获取龙门架的所有记录叫交通事件。且一个 Regional Center 有多个 C2 对象与之对应。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 和问题 2，将解答填入答题纸的对应栏内。

【说明】

某公司购买长钢条，将其切割后进行出售。切割钢条的成本可以忽略不计，钢条的长度为整英寸。已知价格表 P，其中 P_i ($i=1, 2, \dots, m$) 表示长度为 i 英寸的钢条的价格。现要求解使销售收益最大的切割方案。

求解此切割方案的算法基本思想如下：

假设长钢条的长度为 n 英寸，最佳切割方案的最左边切割段长度为 i 英寸，则继续求解剩余长度为 $n-i$ 英寸钢条的最佳切割方案。考虑所有可能的 i ，得到的最大收益 r_n 对应的切割方案即为最佳切割方案。 r_n 的递归定义如下：

$$r_n = \max_{1 \leq i \leq n} (p_i + r_{n-i})$$

对此递归式，给出自顶向下和自底向上两种实现方式

【C 代码】

/*常量和变量说明

n: 长钢条的长度

P[]: 价格数组

*/

#define LEN 100

int Top__Down__ Cut_Rod(int P[], int n) { /*自顶向下*/

Int r=0

Int i;

if(n=0) {

return 0;

}

for(i=1; (1); i++) {

int tmp=p[i]+Top_Down_ Cut_Rod(p, n-i)

r=(r>=tmp)?r: tmp;

}

return r;

}

```

int Bottom_Up_Cut_Road(int p[], int n) { /*自底向上*/

    int r[LEN]={0};

    int temp=0;

    int i, j;

    for(j=1; j<=n; j++) {

        temp=0;

        for(i=1; (2); i++) {

            temp=(3);

        }

        (4)

    }

    return r[n];

}

```

【问题 1】(8 分)

根据说明，填充 C 代码中的空 (1) ~ (4)。

【问题 2】(7 分)

根据说明和 C 代码，算法采用的设计策略为 (5)。求解时，自顶向下方法的时间复杂度为 (6)；自底向上方法的时间复杂度为 (7) (用 0 表示)。

【答案】

问题 1:

(1): $i \leq n$

(2): $i \leq j$

(3): $\text{temp} = (\text{temp} > r[i] + r[j - i]) ? \text{temp} : (r[i] + r[j - i])$

(4): $r[j] = (\text{temp} > p[j]) ? \text{temp} : p[j];$

问题 2:

(5) 动态规划法

(6) $O(2n)$

(7) $O(n^2)$

【解析】

问题 2:

题干中提到说考虑所有可能的 i ，得到最大收益的方式，而自底向上算法实现时，使用到数组把其中最优的解记录。是动态规划法。

从下列的 2 道试题(试题五至试题六)中任选 1 道解答。请在答题纸上的指定位置处将所选试题的题号框涂黑。若多涂或者未涂题号框, 则对题号最小的一道试题进行评分。

试题五(共 15 分)

阅读下列说明和 C++代码, 将应填入 (n) 处的字句写在答题纸的对应栏内。

【说明】

生成器 (Builder) 模式的意图是将一个复杂对象的构建与它的表示分离, 使得同样的构建过程可以创建不同的表示。图 5-1 所示为其类图。

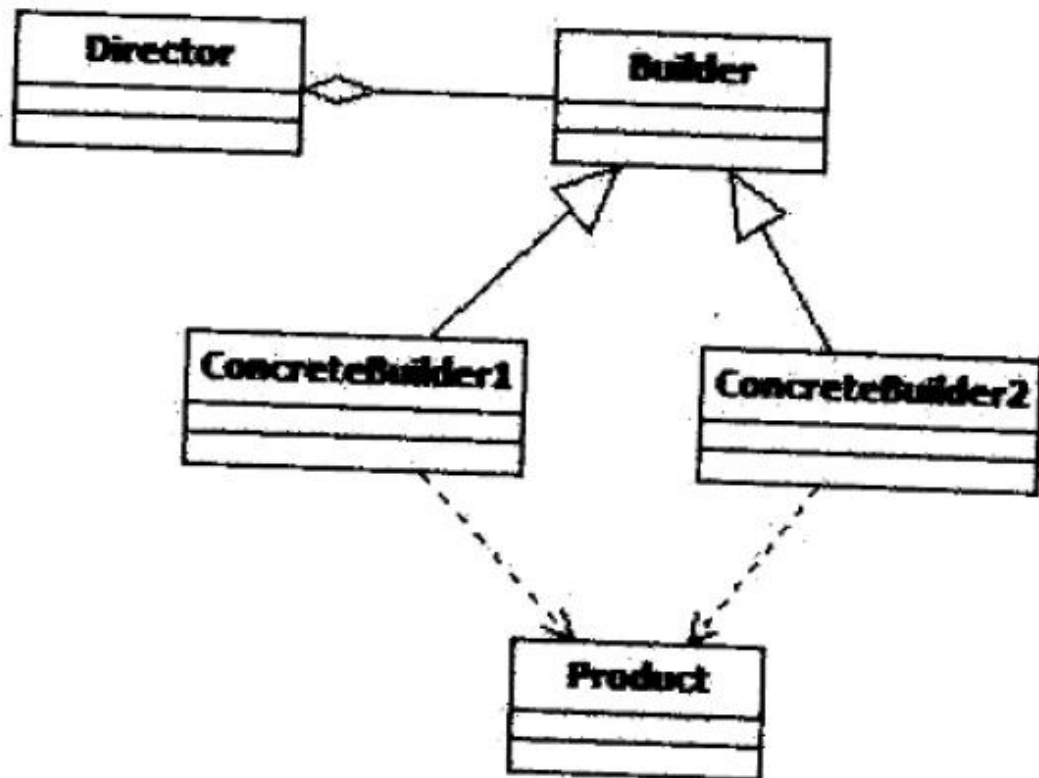


图 5-1 生成器模式类图

【C++代码】

```
#include <iostream>

#include <string>

using namespace std;

class Product {

    private: string partA, partB;

    public: Product() { }
```



```

        void setPartA(const string&s) { PartA=s;}

        void setPartB(const string&s) { PartB=s;}

        // 其余代码省略
};

class Builder{
    public: (1);

    virtual void buildPartB()=0;

    (2);
};

class ConcreteBuilder1: public Builder{
    private: Product* product;

    public: ConcreteBuilder1() {product=new Product(); }

        void buildPartA() { (3) ("Component A");}

        void buildPartB() { (4) ("Component B");}

        Product*getResult() { return product;}

        // 其余代码省略
};

class ConcreteBuilder2: public Builder{
    /*代码省略*/
};

class Director {
    private: Builder* builder;

    public: Director(Builder*pBuilder) {builder*pBuilder;}

        void construct() {

            (5)

            // 其余代码省略

        }

        // 其余代码省略
};

int main() {

```

```
Director* director1=new Director(new ConcreteBuilder1());  
director1->construct();  
delete director 1;  
return 0;  
}
```

【答案】

- (1) `virtual void buildPartA() = 0`
- (2) `virtual Product * getResult() = 0`
- (3) `product->setPartA`
- (4) `product->setPartB`
- (5) `builder->buildPartA();`
`builder->buildPartB();`
`Product* p = builder->getResult();`

【解析】

第 5 空，由于后面的代码省略，所以以上三个都可。

试题六(共 15 分)

阅读下列说明和 Java 代码, 将应填入(n)处的字句写在答题纸的对应栏内。

【说明】

生成器(Builder)模式的意图是将一个复杂对象的构建与它的表示分离, 使得同样的构建过程可以创建不同的表示。图 6-1 所示为其类图。

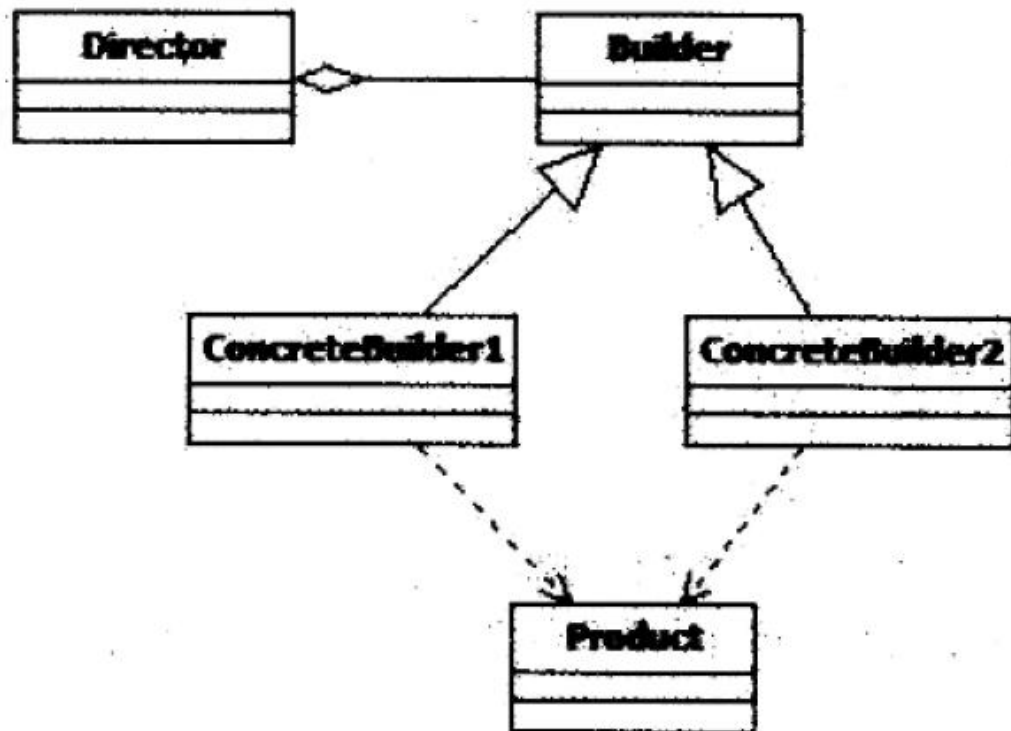


图 6-1 生成器模式类图

【Java 代码】

```
import java.util.*;

class Product {

    private String part A;

    private String part B;

    public Product() {}

    public void setPartA(String s) { partA=s;}

    public void setPartB(String s) { partB=s;}

}
```

```

interface Builder{

    public (1);

    public void buildPartB();

    public (2);

}

class ConcreteBuilder1: implements Builder{

    private Product product;

    public ConcreteBuilder1(){product=new Product(); }

    public void buildPartA(){ (3) ("Component A");}

    public void buildPartB(){ (4) ("Component B");}

    public Product getResult(){ return product;}

}

class ConcreteBuilder2 implements Builder{

    //代码省略

}

class Director {

    private Builder builder;

    public Director(Builder builder){this.builder=builder;}

    public void construct(){

        (5)

        //代码省略

    }

}

class Test{

    public static void main(String[]args){

        Director director1=new Director(new ConcreteBuilder1());

        director1.construct(); }

}

```

【答案】

- (1) `virtual void buildPartA()`
- (2) `virtual Product getResult()`
- (3) `product.setPartA`
- (4) `product.setPartB`
- (5) `builder.buildPartA();`
`builder.buildPartB();`
`Product p = builder.getResult();`

【解析】

第 5 空，由于后面的代码省略，所以以上三个都可。