在 CPU 中，常用来为 ALU 执行算术逻辑运算提供数据并暂存运算结果的寄存器是（1）。

(1)A.程序计数器　　B.状态寄存器　　C.通用寄存器　　D.累加寄存器

【答案】D

【解析】本题考查计算机系统基础知识。

CPU 中有一些重要的寄存器，程序计数器（ PC ）用于存放指令的地址。 当程序顺序执行时，每取出一条指令， PC 内容自动增加一个值，指向下一条要取的指令，当程序出现转移时，则将转移地址送入 PC，然后由 PC 给出新的指令地址。

状态寄行器用于记录运算中产生的标志信息。状态寄存器中的每一位单独使用，成为标志位。标志位的取值反映了 ALU 当前的工作状态，可以作为条件转移指令的转移条件。典型的标志位有以下几种：进位标志位（C）、零标志位（Z）、符号标志位（S）、溢出标志位（V）、奇偶标志位（P）。

通用寄存器组是 CPU 中的一组工作寄存器，运算时用于暂存操作数或地址。在程序中使用通用寄存器可以减少访问内存的次数，提高运算速度。累加器（accumulator）是一个数据寄存器，在运算过程中暂时存放操作数和中间运算结果，不能用于长时间地保存一个数据。

某机器字长为 n，最高位是符号位，其定点整数的最大值为（2）。

(2)A.2n-1　　B.2n-1-1　　C.2n　　D.2n-1

【答案】B

【解析】本题考查计算机系统中的数据表示基础知识。

机器字长为 n，最高位为符号位，则剩余的 n-1 位用来表示数值，其最大值是这 n-1 位都为 1，也就是 2n-1-1。

海明码利用奇偶性检错和纠错，通过在 n 个数据位之间插入 k 个校验位，扩大数据编码的码距。若 n=48，则 k 应为（3）。

(3)A.4　　B.5　　C.6　　D.7

【答案】C

【解析】本题考查数据校验基础知识。

设数据位是 n 位，校验位是 k 位，则 n 和 k 必须满足以下关系：$2k-1 \geqslant n + k$。若 n=48，则 k 为 6 时可满足 $26-1 \geqslant 48+6$ 。

海明码的编码规则如下。

设 k 个校验位为 Pk，Pk-1，…，P1， n 个数据位为 Dn-1，Dn-2，…，D1，D0 。对应的海明码为 Hn+k，Hn+k-1，…，H1，那么：

①Pi 在海明码的第 2i-1 位置，即 Hj=Pi，且 j＝2i-1；数据位则依序从低到高占据海明码中剩下的位置。

②海明码中的任一位都是由若干个校验位来校验的。其对应关系如下：被校验的海明位的下标等于所有参与校验该位的校验位的下标之和，而校验位则由自身校验。

通常可以将计算机系统中执行一条指令的过程分为取指令，分析和执行指令 3 步。若取指令时间为 4△t，分析时间为 2△t，执行时间为 3△t，按顺序方式从头到尾执行完 600 条指令所需时间为（4）△t；若按照执行第 i 条，分析第 i+1 条，读取第 i+2 条重叠的流水线方式执行指令，则从头到尾执行完 600 条指令所需时间为（5）△t 。

(4)A.2400　　B.3000　　C.3600　　D.5400

(5)A.2400　　B.2405　　C.3000　　D.3009

【答案】D　B

【解析】本题考查指令系统基础知识。

(4)指令顺序执行时，每条指令需要 9△t（4△t+2△t+3△t），执行完 600 条指令需要 5400△t。

(5) 若采用流水方式，则在分析和执行第 1 条指令时，就可以读取第 2 条指令，当第 1 条指令执行完成，第 2 条指令进行分析和执行，而第 3 条指令可进行读取操作。因此，第 1 条指令执行完成后，每 4△t 就可以完成 1 条指令，600 条指令的总执行时间为 9△t +599×4△t = 2405△t 。

若用 256K×8bit 的存储器芯片，构成地址 40000000H 到 400FFFFFH 且按字节编址的内存区域，则需（6）片芯片。

(6)A.4　　B.8　　C.16　　D.32

【答案】A

【解析】本题考查计算机系统中存储器知识。

地址 400000000H 到 4000FFFFFH 共 FFFFFH （即 220 ）个以字节为单位的编址单元，而 256K ×8bit 的存储器芯片可提供 218 个以字节为单位的编址单元， 因此需要 4 片（220/218 ）这种芯片来构成上述内存区域。

以下关于木马程序的叙述中，正确的是（7）。

(7)A.木马程序主要通过移动磁盘传播

B.木马程序的客户端运行在攻击者的机器上

C.木马程序的目的是使计算机或网络无法提供正常的服务

D.Sniffer 是典型的木马程序

【答案】B

【解析】本题考查木马程序的基础知识。

木马程序一般分为服务器端（Server）和客户端（Client），服务器端是攻击者传到目标机器上的部分，用来在目标机上监听等待客户端连接过来。客户端是用来控制目标机器的部分，放在攻击者的机器上。

木马（Trojans）程序常被伪装成工具程序或游戏，一旦用户打开了带有特洛伊木马程序的邮件附件或从网上直接下载，或执行了这些程序之后，当你连接到互联网上时，这个程序就会通知黑客用户的 P 地址及被预先设定的端口。黑客在收到这些资料后，再利用这个潜伏其中的程序，就可以恣意修改用户的计算机设定、复制任何文件、窥视用 户整个硬盘内的资料等，从而达到控制用户的计算机的目的。

现在有许多这样的程序，国外的此类软件有 Back Office、Netbus 等，国内的此类软件有 Netspy、YAI、SubSeven、冰河、“广外女生”等。Sniffer 是一种基于被动侦听原理的网络分析软件。使用这种软件，可以监视网络的状态、数据流动情况以及网络上传输的信息，其不属于木马程序。

防火墙的工作层次是决定防火墙效率及安全的主要因素，以下叙述中，正确的是（8）。

(8)A.防火墙工作层次越低，工作效率越高，安全性越高

B.防火墙工作层次越低，工作效率越低，安全性越低

C.防火墙工作层次越高，工作效率越高，安全性越低

D.防火墙工作层次越高，工作效率越低，安全性越高

【答案】D

【解析】本题考查防火墙的基础知识。

防火墙的性能及特点主要由以下两方面所决定。

①工作层次。这是决定防火墙效率及安全的主要因素。一般来说，工作层次越低，则工作效率越高，但安全性就低了；反之，工作层次越高，工作效率越低，则安全性越高。

②防火墙采用的机制。 如果采用代理机制， 则防火墙具有内部信息隐藏的特点， 相对而言， 安全性高， 效率低；如果采用过滤机制， 则效率高， 安全性却降低了。

以下关于包过滤防火墙和代理服务防火墙的叙述中，正确的是（9）。

(9)A.包过滤成本技术实现成本较高，所以安全性能高

B.包过滤技术对应用和用户是透明的

C.代理服务技术安全性较高，可以提高网络整体性能

D.代理服务技术只能配置成用户认证后才建立连接

【答案】B

【解析】本题考查防火墙的基础知识。

显然， 包过滤防火墙采用包过滤技术对应用和用户是透明的。

王某买了一幅美术作品原件，则他享有该美术作品的(10)。

(10)A.著作权　　B.所有权　　C.展览权　　D.所有权与其展览权

【答案】D

【解析】本题考查知识产权基本知识。

绘画、书法、雕塑等美术作品的原件可以买卖、赠与。 但获得一件美术作品并不意味着获得该作品的著作权。我国著作权法规定：“美术等作品原件所有权的转移， 不视为作品著作权的转移，但美术作品原件的展览权由原件所有人享有。”这就是说作品物转移的事实并不引起作品著作权的转移， 受让人只是取得物的所有权和作品原件的展览权， 作品的著作权仍然由作者享有。

甲、乙两软件公司于 2012 年 7 月 12 日就其财务软件产品分别申请“用友”和“用有”商标注册。两财务软件相似，甲第一次使用时间为 2009 年 7 月，乙第一次使用时间为 2009 年 5 月。此情形下，（11）能获准注册。

(11)A.“用友”　　B.“用友”与“用有”都

C.“用有”　　D.由甲、乙抽签结果确定谁

【答案】C

【解析】

《中华人民共和国商标法》第二十九条　两个或者两个以上的商标注册申请人，在同一

种商品或者类似商品上，以相同或者近似的商标申请注册的，初步审定并公告申请在先的商标；同一天申请的，初步审定并公告使用在先的商标，驳回其他人的申请，不予公告。

《中华人民共和国商标法实施条例》第十九条 两个或者两个以上的申请人，在同一种商品或者类似商品上，分别以相同或者近似的商标在同一天申请注册的，各申请人应当自收到商标局通知之日起30日内提交其申请注册前在先使用该商标的证据。同日使用或者均未使用的，各申请人可以自收到商标局通知之日起30日内自行协商，并将书面协议报送商标局；不愿协商或者协商不成的，商标局通知各申请人以抽签的方式确定一个申请人，驳回其他人的注册申请。商标局已经通知但申请人未参加抽签的，视为放弃申请，商标局应当书面通知未参加抽签的申请人。

所以，同日申请选择先使用的，即“用有”。

以下媒体中，（12）是表示媒体，（13）是表现媒体。

(12)A.图像　　B.图像编码　　C.电磁波　　D.鼠标

(13)A.图像　　B.图像编码　　C.电磁波　　D.鼠标

【答案】B　D

【解析】本题考查多媒体基础知识。

国际电话电报咨询委员会（CCITT）将媒体分为感觉媒体、 表示媒体、 表现媒体、存储媒体和传输媒体5类。

感觉媒体指直接作用于人的感觉器官.，使人严生内接感觉的媒体， 如引起听觉反应的声音、引起视觉反应的图像等；

传输媒体指传输表示媒体的物理介质，如电缆、光缆。电磁波等；

表示媒体指传输感觉媒体的中介媒体，即用于数据交换的编码，如图像编码、文本编码和声音编码等；

表现媒体是指进行信息输入和输出的媒体，如键盘、鼠标、话筒，以及显示器、打印机、喇叭等；

存储媒体指用于存储表示媒体的物理介质，如硬盘、光盘等。

(14)表示显示器在横向（行）上具有的像素点数目。

(14)A.显示分辨率　　B.水平分辨率　　C.垂直分辨率　　D.显示深度

【答案】B

【解析】本题考查多媒体基础知识。

显示分辨率是指显示器上能够显示出的像素点数目，即显示器在横向和纵向上能够显示出的像素点数目。水平分辨率表明显示器水平方向（横向）上显示出的像素点数目，垂直分辨率表明显示器垂直方向（纵向）上显示出的像素点数目。例如，显示分辨率为 1024X768 则表明显示器水平方向上显示 1024 个像素点，垂直方向上显示 768 个像素点，整个显示屏就含有 796432 个像素点。屏幕能够显示的像素越多，说明显示设备的分辨率越高，显示的图像质量越高。显示深度是指显示器上显示每个像素点颜色的二进制位数。

以下关于结构化开发方法的叙述中，不正确的是(15)。

(15)A.将数据流映射为软件系统的模块结构

B.一般情况下，数据流类型包括变换流型和事务流型

C.不同类型的数据流有不同的映射方法

D.一个软件系统只有一种数据流类型

【答案】D

【解析】本题考查结构化开发方法的结构化设计。

结构化设计方法是一种面向数据流的设计方法，与结构化分析方法衔接。在需求分析阶段，结构化分析方法产生了数据流图，而在设计阶段，结构化设计方法将数据流映射为软件系统的模块结构。数据流图中从系统的输入数据流到系统的输出数据流的一连串变换形成了一条信息流。其中的信息流一般情况下包括变换流型和事物流型不同类型的数据流到程序模块的映射方法不同。--一个软件系统往往不仅仅有一种数据流类型。

模块 A 提供某个班级某门课程的成绩给模块 B，模块 B 计算平均成绩、最高分和最低分，将计算结果返回给模块 A ，则模块 B 在软件结构图中属于（16）模块。

(16)A.传入　　B.传出　　C.变换　　D.协调

【答案】C

【解析】本题考查结构化开发方法的基础知识。

通常，可以按照在软件系统中的功能将模块分为四种类型。①传入模块:取得数据或输入数据，经过某些处理，再将其传送给其他模块.②传出模块：输出数据，在输出 之前可能进行某些处理，数据可能被输出到系统的外部，或者会输出到其他模块进行进一步处理。③

变换模块：从上级调用模块得到数据，进行特定的处理，转换成其他形式，在将加工结果返回给调用模块。④协调模块一般不对数据进行加工，主要是通过调用、协调和管理其他模块来完成特定的功能。

(17) 软件成本估算模型是一种静态单变量模型，用于对整个软件系统进行估算。

(17)A.Putnam　　B.基本 COCOMO　　C.中级 COCOMO　　D.详细 COCOMO

【答案】B

【解析】本题考查软件项目管理的基础知识。

Putnam 和 COCOMO 都是软件成本估算模型。Putnam 模型是一种动态多变量模型，假设在软件开发的整个生存期中工作量有特定的分布。结构性成本模型 COCOMO 模型分为基本 COCOMO 模型、中级 COCOMO 模型和详细 COCOMO。基本 COCOMO 模型是一个静态单变量模型，对整个软件系统进行估算；中级 COCOMO 模型是一个静态多变量模型，将软件系统模型分为系统和部件两个层次，系统由部件构成；详细 COCOMO 模型将软件系统模型分为系统、子系统和模块三个层次，除了包括中级模型所考虑的因素外，还考虑了在需求分析、软件设计等每一步的成本驱动属性的影响。

以下关于进度管理工具 Gantt 图的叙述中，不正确的是(18)。

(18)A.能清晰地表达每个任务的开始时间、结束时间和持续时间

B.能清晰地表达任务之间的并行关系

C.不能清晰地确定任务之间的依赖关系

D.能清晰地确定影响进度的关键任务

【答案】D

【解析】本题考查软件项目管理的基础知识。

Gantt 图是一种简单的水平条形图，以日历为基准描述项目任务。水平轴表示日历时间线，如日、周和月等，每个条形表示一个任务，任务名称垂直的列在左边的列中，图中水平条的起点和终点对应水平轴上的时间，分别表示该任务的开始时间和结束时间，水平条的长度表示完成该任务所持续的时间。当日历中同一时段存在多个水平条时，表示任务之间的并发。

Gantt 图能清晰地描述每个任务从何时开始，到何时结束，任务的进展情况以及各个任务之间的并行性。但它不能清晰地反映出各任务之间的依赖关系，难以确定整个项目的关键

所在，也不能反映计划中有潜力的部分。

项目复杂性、规模和结构的不确定性属于(19)风险。

(19)A.项目　　B.技术　　C.经济　　D.商业

【答案】A

【解析】本题考查软件项目管理的基础知识。

项目经理需要尽早预测项目中的风险，这样就可以制订有效的风险管理计划以减少风险的影响，所以，早期的风险识别是非常重要的，一般来说，影响软件项目的风险主要有三种类别：项目风险涉及到各种形式的预算、进度、人员、资源以及和客户相关的问题；技术风险涉及到潜在的设计、实现、对接、测试即维护问题；业务风险组括建立一个无人想要的优秀产品的风险、失去预算或人员承诺的风险等：商业风险包括如市场风险、策略风险、管理风险和预算风险等.

以下程序设计语言中，(20) 更适合用来进行动态网页处理。

(20)A.HTML　　B.LISP　　C.PHP　　D.JAVA/C++

【答案】C

【解析】本题考查程序语言基础知识。

网页文件本身是一种文本文件，通过在其中添加标记符，可以告诉浏览器如何显示其中的内容。HTML 是超文本标记语言，超文本是指页面内可以包含图片、链接，甚至 音乐、程序等非文字元素。

PHP（超文本预处理器）是一种通用开源脚本语言，它将程序嵌入到 HTML 文档中去执行，从而产生动态网页。

在引用调用方式下进行函数调用是将(21)。

(21)A.实参的值传递给形参　　B.实参的地址传递给形参

C.形参的值传递给实参　　D.形参的地址传递给实参

【答案】B

【解析】本题考查程序语言基础知识。

值调用和引用调用是实现函数调用时传递参数的两种基本方式。在值调用方式下，是将实参的值传给形参，在引用调用方式下，是将实参的地址传递给形参。

编译程序对高级语言源程序进行编译的过程中，要不断收集、记录和使用源程序中一些相关符号的类型和特征等信息，并将其存入(22)中。

(22)A.符号表　　B.哈希表　　C.动态查找表　　D.栈和队列

【答案】A

【解析】本题考查程序语言基础知识。

编译是实现高级程序设计语言的一种方式，编译过程可分为词法分析、语法分析、语义分析、中间代码生成、代码优化和目标代码生成等阶段，还需要进行出错处理和符号表管理。符号表的作用是记录源程序中各个符号的必要信息，以辅助语义的正确性检查和代码生成，在编译过程中需要对符号表进行快速有效地查找、插入、修改和删除等操作符号表的建立可以始于词法分析阶段，也可以放到语法分析和语义分析阶段，但符号表的使用有时会延续到目标代码的运行阶段。

设计操作系统时不需要考虑的问题是(23)。

(23)A.计算机系统中硬件资源的管理　　B.计算机系统中软件资源的管理

C.用户与计算机之间的接口　　D.语言编译器的设计实现

【答案】D

【解析】

操作系统设计的目的是管理计算机系统中的软硬件资源，为用户与计算机之间提供方便的接口。

假设某计算机系统中资源 R 的可用数为 6，系统中有 3 个进程竞争 R，且每个进程都需要 i 个 R，该系统可能会发生死锁的最小 i 值是(24)。若信号量 S 的当前值为-2，则 R 的可用数和等待 R 的进程数分别为(25)。

(24)A.1　　B.2　　C.3　　D.4

(25)A.0、0　　B.0、1　　C.1、0　　D.0、2

【答案】C　D

【解析】本题考查对操作系统进程管理信号量方面的基础知识。

(24)选项 A 是错误的，因为每个进程都需要 1 个资源 R，系统为 3 个进程各分配 1 个，系统中资源 R 的可用数为 3，3 个进程都能得到所需资源，故不发生死锁；选项 B 是错

误的，因为，每个进程都需要 2 个资源 R，系统为 3 个进程各分配 2 个，系统中资源 R 的可用数为 6，3 个进程都能得到所需资源，故也不发生死锁；选项 C 是正确的，因为，每个进程都需要 3 个资源 R，系统为 3 个进程各分配 2 个，系统中资源 R 的可用数为 6，3 个进程再申请 1 个资源 R 得不到满足，故发生死锁；选项 D 显然是错误的。

(25) 早在 1965 年荷兰学者 Dijkstra 提出信号量机制是一种有效的进程同步与互斥工具。目前，信号量机制有了很大的发展，主要有整型信号量、记录型信号量和信号量集机制。对于整型信号量，可以根据控制对象的不同被赋予不同的值。通常将信号量分为公用信号量和私用信号量两类。其中，公用信号量用于实现进程间的互斥，初值为 1 或资源的数目；私用信号量用于实现进程间的同步，初值为 0 或某个正整数。信号量 S 的物理意义是：S≥0 表示某资源的可用数，若 S<0,则其绝对值表示阻塞队列中等待该资源的进程数。本题由于信号量 S 的当前值为-2,则意味着系统中资源 R 的可用个数 M=0，等待资源 R 的进程数 N=2。

某计算机系统页面大小为 4K，若进程的页面变换表如下所示，逻辑地址为十六进制 1D16H。该地址经过变换后，其物理地址应为十六进制(26)。

页号	物理块号
0	1
1	3
2	4
3	6

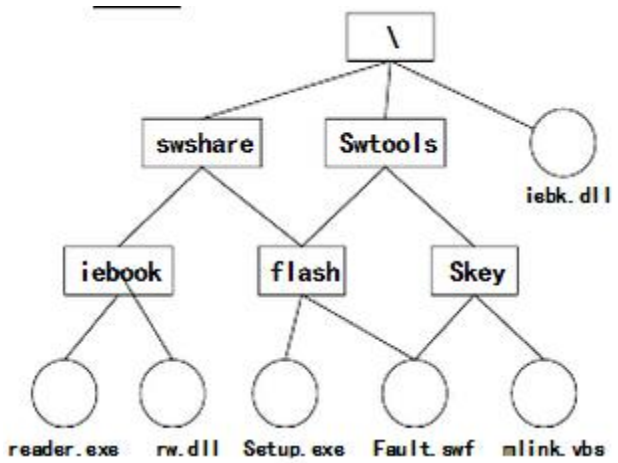
(26)A.1024H　　B.3D16H　　C.4D16H　　D.6D16H

【答案】B

【解析】

根据题意页面大小为 4K，逻辑地址为十六进制 1D16H 其页号为 1，页内地址为 D16H,查页表后可知物理块号为 3,该地址经过变换后，其物理地址应为物理块号 3 拼上页内地址 D16H ，即十六进制 3D16H。

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 fault.swf，且当前工作目录为 swshare，则该文件的全文件名为(27)，相对路径和绝对路径分别为(28)。



(27)A.fault.swf　　B.flash\fault.swf

C.swshare\flash\fault.swf　　D.\swshare\flash\fault.swf

(28)A.swshare\flash\和\flash　　B.flash\和\swshare\flash

C.\swshare\flash\和\flash　　D.\flash\和\swshare\flash

【答案】D　B

【解析】本题考查对操作系统文件管理方面的基础知识。

路径名是指操作系统查找文件所经过的目录名以及目录名之间的分隔符构成的。通常，操作系统中全文件名是指路径名+文件名。

按查找文件的起点不同可以将路径分为：绝对路径和相对路径。从根目录开始的路径称为绝对路径；从用户当前工作目录开始的路径称为相对路径，相对路径是随着当前工作目录的变化而改变的。

以下关于统一过程 UP 的叙述中，不正确的是(29)。

(29)A.UP 是以用例和风险为驱动，以架构为中心，迭代并且增量的开发过程

B.UP 定义了四个阶段，即起始、精化、构建和确认阶段

C.每次迭代都包含计划、分析、设计、构造、集成、测试以及内部和外部发布

D.每个迭代有五个核心工作流

【答案】B

【解析】本题考查软件过程模型的基础知识。

UP（统一过程）模型是一种以用例和风险为驱动、以架构为中心、迭代并且增量的开发过程，由 UML 方法和工具支持。UP 过程定义了五个阶段，起始阶段、精化阶段、构建阶段、移交阶段和产生阶段。开发过程中有多次迭代，每次迭代都包含计划、分析、 设计、构造、

集成和测试，以及内部和外部发布。每个迭代有五个核心工作流，捕获系统应该做什么的需求工作流、精化和结构化需求的分析工作流、在系统结构内实现需求的设计工作流、构造软件的实现工作流和验证是否如期望那样工作的测试工作流。

某公司要开发一个软件产品，产品的某些需求是明确的，而某些需求则需要进一步细化。由于市场竞争的压力，产品需要尽快上市，则开发该软件产品最不适合采用(30)模型。

(30)A.瀑布　　B.原型　　C.增量　　D.螺旋

【答案】A

【解析】本题考查软件过程模型的基础知识。

瀑布模型将软件生存周期各个活动规定为线性顺序连接的若干阶段的模型，规定了由前至后，相互衔接的固定次序，如同瀑布流水，逐级下落。这种方法是一种理想的开发模式，缺乏灵活性，特别是无法解决软件需求不明确或不准确的问题。原型模型从初始的原型逐步演化成最终软件产品，特别适用于对软件需求缺乏准确认识的情况。螺旋将瀑布模型与快速原型模型结合起来，并且加入两种模型均忽略了的风险分析，适用于复杂的大型软件。增量是开发把软件产品作为一系列的增量构件来设计、编码、集成和测试，可以在增最开发过程中逐步理解需求。

在屏蔽软件错误的容错系统中，冗余附加技术的构成不包括(31)。

(31)A.关键程序和数据的冗余存储及调用　　B.冗余备份程序的存储及调用

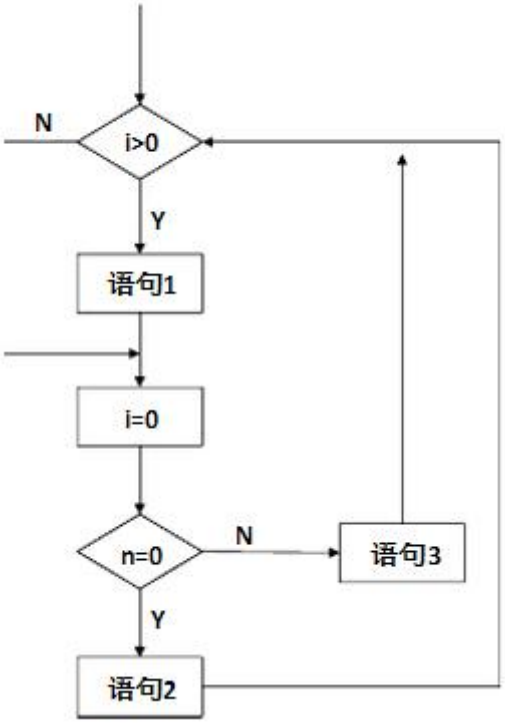
C.实现错误检测和错误恢复的程序　　D.实现容错软件所需的固化程序

【答案】A

【解析】本题考查软件容错技术的基础知识。

容错技术是对某些无法避开的差错，使其影响减至最小的技术。通常冗余技术分为四类，结构冗余、信息冗余、时间冗余和冗余附加技术。其中冗余附加技术是指为实现其他类型冗余技术所需要的资源和技术，包括程序指令、数据、存放和调动它们的空间和通道等。在屏蔽硬件错误的容错技术中，冗余附加技术包括：关键程序和数据的冗余存储及调用：检测、表决、切换、重构、纠错和复算的实现。在屏蔽软件错误的容错技术中，冗余附加技术包括：冗余备份程序的存储及调用；实现错误检测和错误恢复的程序；实现容错软件所需的固化程序。

采用 McCabe 度量法计算下列程序图的环路复杂性为(32)。



(32)A.2　　B.3　　C.4　　D.5

【答案】C

【解析】本题考查软件质量的基础知识。

McCabe 度量法是一种基于程序控制流的复杂性度量方法，环路复杂性为 V(G) = m-n + 2，图中 m = 8，n = 6，V(G) = 8-6 + 2 = 4。

以下关于文档的叙述中，不正确的是(33)。

(33)A.文档仅仅描述和规定了软件的使用范围及相关的操作命令

B.文档也是软件产品的一部分，没有文档的软件就不能称之为软件

C.软件文档的编制在软件开发工作中占有突出的地位和相当大的工作量

D.高质量文档对于发挥软件产品的效益有着重要的意义

【答案】A

【解析】本题考查软件文档的基础知识。

软件文档不仅仅描述和规定了软件的使用范围及相关的操作命令，还包括硬件采购和网络设计中形成的文档。文档是软件产品的重要组成部分。软件文档的编制在软件开发工作中占有突出的地位和相当大的工作量，对发挥软件产品的效益具有重要的意义。

某搜索引擎在使用过程中，若要增加接受语音输入的功能，使得用户可以通过语音输入来进行搜索，此时应对系统进行(34)维护。

(34)A.正确性　　B.适应性　　C.完善性　　D.预防性

【答案】B

【解析】本题考查软件维护的基础知识。

软件维护一般包括四种类型：

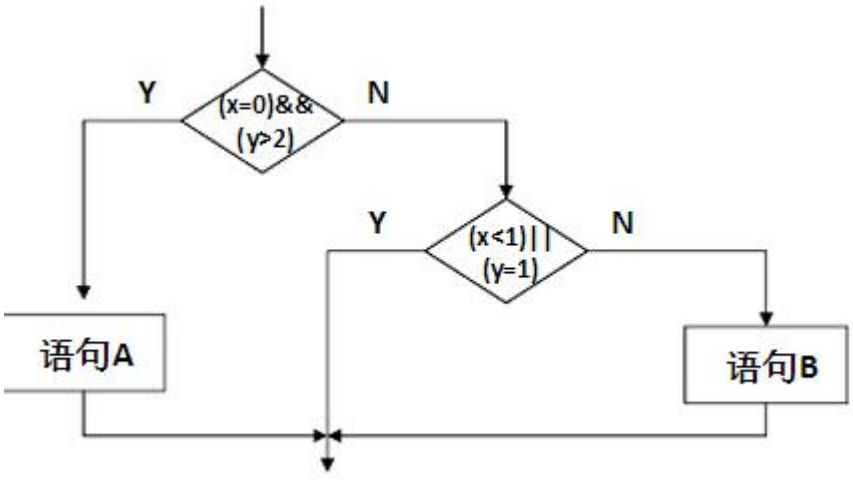
正确性维护是指改正在系统开发阶段已发生而系统测试阶段尚未发现的错误；

适应性维护是指使应用软件适应新型技术变化和管理需求变化而进行的修改；

完善性维护是指为扩充功能和改善性能而进行的修改，主要是指对已有的软件系统增加一些在系统分析和设计阶段中没有规定的功能与性能特征；

预防性维护是指为了改进应用软件的可靠性和可维护性，为了适应未来的软硬件环境的变化，主动增加预防性的功能，以使应用系统适应各类变化而不被淘汰。

采用白盒测试方法对下图进行测试，设计了 4 个测试用例：①(x=0，y=3)，②(x=1, y=2)，③(x=-1，y=2)，④(x=3, y=1)。至少需要测试用例①②才能完成(35)覆盖，至少需要测试用例①②③或①②④才能完成(36)覆盖。



(35)A.语句　　B.条件　　C.判定 / 条件　　D.路径

(36)A.语句　　B.条件　　C.判定 / 条件　　D.路径

【答案】A　D

【解析】本题考查软件测试的基础知识。

测试用例①（x=0，y=3)在第一个判断结果为 Y，执行语句 A;测试用例②（x=1，y=2） 在第一个判断结果为 N，第二个判断结果为 N,执行语句 B;测试用例③(x=-1, y=2) 和④（x=3,

y=1）在第一个判断结果为N,第二个判断结果为Y。至少需要测试用例① ②才能完成语句覆盖，至少需要测试用例①②③或①②④才能完成路径覆盖。

(37)是一个类与它的一个或多个细化类之间的关系，即一般与特殊的关系。

(37)A.泛化　　B.关联　　C.聚集　　D.组合

【答案】A

【解析】本题考查面向对象的基本知识。

泛化是一个类与它的一个或多个细化类之间的关系，表达一般与特殊的关系。关联是类与类之间的一种结构关系。聚集是一种关系，其中一个较大的整体类包含一个或多个较小的部分类；相反地，一个较小的部分类是一个较大的整体类的一部分。组合是一种聚合关系，其中整体负责其部分的创建和销毁，如果整体不存在了，部分也将不存在。

某些程序设计语言中，在运行过程中当一个对象发送消息请求服务时，根据接收对象的具体情况将请求的操作与实现的方法进行连接，称为(38)。

(38)A.静态绑定　　B.通用绑定　　C.动态绑定　　D.过载绑定

【答案】C

【解析】本题考查面向对象的基本知识。

绑定是一个把过程调用和响应调用所需要执行的代码加以结合的过程。在一般的程序设计语言中，绑定是在编译时进行的，叫做静态绑定。动态绑定则是在运行时进行的， 因此，一个给定的过程调用和代码的结合直到调用发生时才进行。

在面向对象技术中，不同的对象在收到同一消息时可以产生完全不同的结果，这一现象称为（39），它由（40）机制来支持。利用类的层次关系，把具有通用功能的消息存放在高层次，而不同的实现这一功能的行为放在较低层次，在这些低层次上生成的对象能够给通用消息以不同的响应。

(39)A.绑定　　B.继承　　C.消息　　D.多态

(40)A.绑定　　B.继承　　C.消息　　D.多态

【答案】D　B

【解析】本题考查面向对象的基本知识。

面向对象技术中，继承关系是一种模仿现实世界中继承关系的一种类之间的关系，是超

类（父类）和子类之间共享数据和方法的机制。父类定义公共的属性和操作，一个父类可以有多个子类，即多个特例。子类可以继承其父类或祖先类中的属性和操作作为自己的内容而不必自己定义，也可以覆盖这些操作，并加入新的内容。绑定是一个把过程调用和响应调用所需要执行的代码加以结合的过程。绑定有在编译时进行的，即静态绑定，有在运行时进行的，即动态绑定。不同的对象收到同一消息可以进行不同的响应，产生完全不同的结果，用户可以发送一个通用的消息，而实现细节则由接收对象自行决 定，使得同一个消息就可以调用不同的方法，即一个对象具有多种形态，称为多态。不同类的对象通过消息相互通信。

对一个复杂用例中的业务处理流程进行进一步建模的最佳工具是 UML(41)。

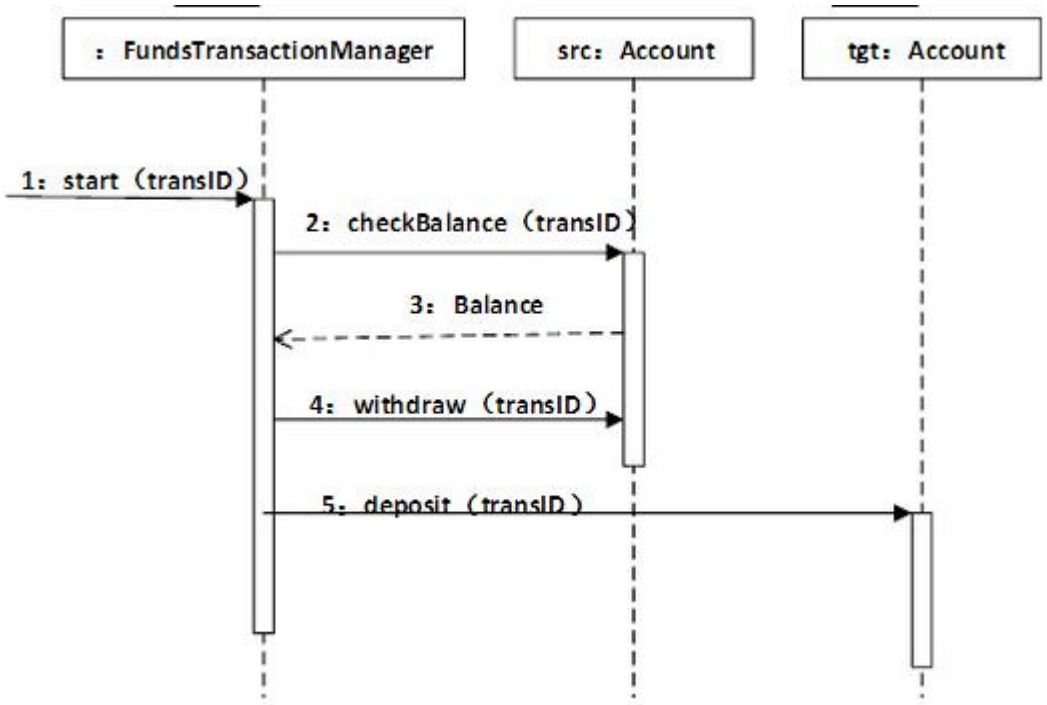
(41)A.状态图　　B.顺序图　　C.类图　　D.活动图

【答案】D

【解析】本题考查采用统一建模语言（UML)的基本知识。

采用 UML 对系统进行建模时，首先确定系统边界，识别出主要用例，建模用例图。然后对用例图中的复杂用例采用活动图进一步进行建模，以对用例中执行过程中对象如何通过消息相互交互进行建模。系统的领域模型采用类图进行建模，交互关系采用顺序图、交互概览图等进行建模。

如下所示的 UML 序列图中，(42)表示返回消息，Account 类必须实现的方法有(43)。



(42)A. tansID　　B. balance　　C. withdraw　　D. deposit

(43)A. start0　　B. checkBalance（）和 withdraw（）

C. deposit0　　D. checkBalance（）、 withdraw（）和 deposit（）

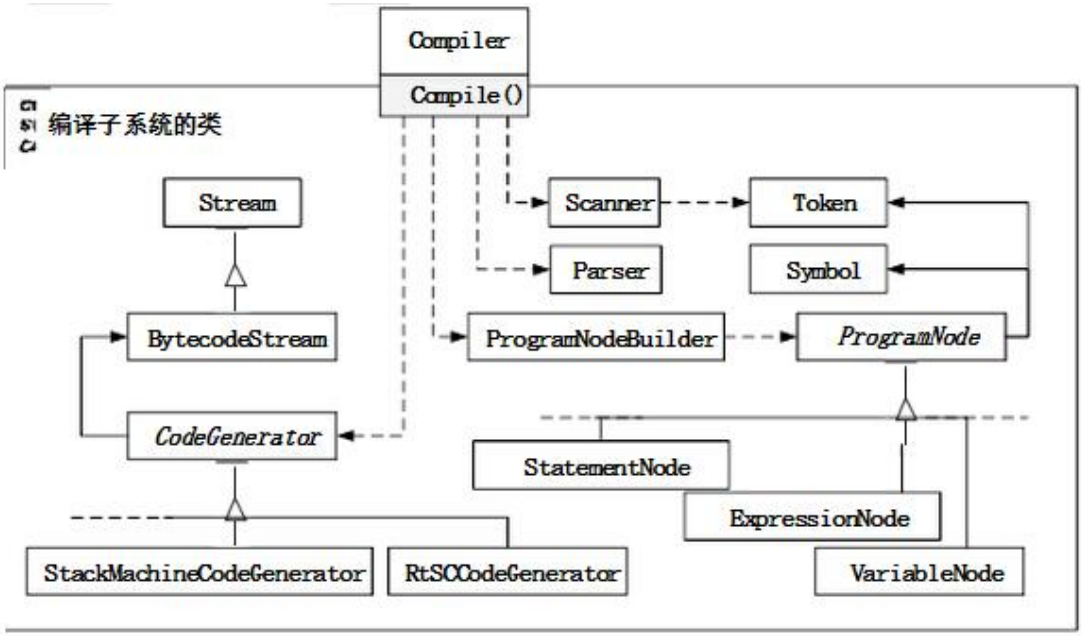
【答案】B　D

【解析】本题考查采用统一建模语言（UML)的基本知识。

UML 序列图（Sequence Diagram)以二维图的形式显示对象之间交互，纵轴自上而下表示时间，横轴表示要交互的对象，主要体现对象间消息传递的时间顺序，强调参与交互的对象及其间消息交互的时序。序列图中包括的建模元素主要有：活动者(Actor)、对象(Object)、生命线（Lifeline)、控制焦点(Focus of control)和消息(Message)等。 其中对象名标有下划线；生命线表示为虚线，沿竖线向下延伸；消息在序列图中标记为箭头；控制焦点由薄矩形表示。

消息是从一个对象的生命线到另一个对象生命线的箭头，用从上而下的时间顺序来安排。一般分为同步消息，异步消息和返回消息。本题图中 balance 为返回消息，其他为同步消息。src 和 tgt 均为 Account 对象，所以 Account 应该实现 checkBalance()、withdraw()和 deposit()方法，FundsTransactionManager 应该实现 start()方法。

下图所示为(44)设计模式，适用于(45)。



(44)A. 适配器(Adapter)　　B. 责任链(Chain of Responsibility)

C. 外观(Facade)　　D. 桥接(Bridge)

(45)A.有多个对象可以处理一个请求，在运行时刻自动确定由哪个对象处理

B.想使用一个已经存在的类，而其接口不符合要求

C.类的抽象和其实现之间不希望有一个固定的绑定关系

D.需要为一个复杂子系统提供一个简单接口

【答案】C　D

【解析】本题考查设计模式的基本概念。

每种设计模式都有特定的意图和适用场景。

适配器(Adapter)模式将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。适用于想使用一个已经存在的类，而其接口不符合要求的情况。

责任链（Chain of Responsibility)模式使多个对象都有机会处理请求，从而避免请求的发送者和接收者之间的耦合关系，将这些对象连成一条链，并沿着这条链传递该请求，直到有一个对象处理它为止。适用于有多个的对象可以处理一个请求，哪个对象处理该请求运行时刻自动确定的情况。

桥接（Bridge)模式将抽象部分与其实现部分分离，使它们都可以独立地变化。适用于不希望在抽象和它的实现部分之间有一个固定的绑定关系的情况。

外观（Facade)模式为子系统中的一组接口提供一个一致的界面，Facade 模式定义了一个高层接口，这个接口使得这一子系统更加容易使用。适用于需要为一个复杂子系统提供一个简单接口的情况。

下列设计模式中，(46)模式既是类结构型模式，又是对象结构型模式。此模式与(47)模式类似的特征是，都给另一个对象提供了一定程度上的间接性，都涉及到从自身以外的一个接口向这个对象转发请求。

(46)A.桥接(Bridge)　　B.适配器(Adapter)

C.组成(Composite)　　D.装饰器(Decorator)

(47)A.桥接(Bridge)　　B.适配器(Adapter)

C.组成(Composite)　　D.装饰器(Decorator)

【答案】B　A

【解析】本题考查设计模式的基本概念。

每种设计模式都有特定的意图，描述一个在我们周围不断重复发生的问题，以及该问题

的解决方案的核心，使该方案能够重用而不必做重复劳动。

适配器(Adapter)模式将一个类或对象的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。既是类结构模式，又是对象结构模式。桥接（Bridge)模式将抽象部分与其实现部分分离，使它们都可以独立地变化。适配器模式和桥接模式具有类似的特征，都给另一个对象提供了一定程度上的间接性，都涉及到自身以外的一个接口向这个对象转发请求。

组合（Composite)模式将对象组合成树形结构以表示“部分-整体”的层次结构， 使得用户对单个对象和组合对象的使用具有一致性。装饰器(Decorator)模式描述了以透明围栏来支持修饰的类和对象的关系，动态地给一个对象添加一些额外的职责，从增加功能的角度来看，装饰器模式相比生成子类更加灵活。

以下关于实现高级程序设计语言的编译和解释方式的叙述中，正确的是(48)。

(48)A.在编译方式下产生源程序的目标程序，在解释方式下不产生

B.在解释方式下产生源程序的目标程序，在编译方式下不产生

C.编译和解释方式都产生源程序的目标程序，差别是优化效率不同

D.编译和解释方式都不产生源程序的目标程序，差别在是否优化

【答案】A

【解析】本题考查程序语言基础知识。

用某种高级语言或汇编语言编写的程序称为源程序，源程序不能直接在计算机上执行。如果源程序是用汇编语言编写的,则需要一个称为汇编程序的翻译程序将其翻泽成目标程序后才能执行。如果源程序是用某种高级语言编写的，则需要对应的解释程序或编译程序对其进行翻译，然后在机器上运行。

解释程序也称为解释器，它可以直接解释执行源程序，或者将源程序翻译成某种中间表示形式后再加以执行；而编译程序（编译器）则首先将源程序翻译成目标语言程序，然后在计算机上运行目标程序。这两种语言处理程序的根本区别是：在编译方式下，机器上运行的是与源程序等价的目标程序，源程序和编译程序都不再参与目标程序的执行过程；而在解释方式下，解释程序和源程序（或其某种等价表示）要参与到程序的运行过程中，运行程序的控制权在解释程序。解释器翻译源程序时不产生独立的目标程序，而编译器则需将源程序翻译成独立的目标程序。

大多数程序设计语言的语法规则用(49)描述即可。

(49)A.正规文法　　B.上下文无关文法　　C.上下文有关文法　　D.短语结构文法

【答案】B

【解析】本题考查程序语言基础知识。

乔姆斯基文法体系共分为短语结构文法、上下文有关文法、上下文有关文法和正规文法4类。

短语结构文法或无限制文法也称为0型文法，其描述能力相当于图灵机，可使用任何的语法描述形式。

上下文有关文法也称为1型文法，其描述能力相当于线性有界自动机，语法形式为：xSy->xAy。也就是说，S（非终结符号）推导出A（非终结符号与终结符号的混合串）是和上下文x，y相关的，即S只有在上下文x，y的环境中才能推导出A。

上下文无关文法也称为2型文法，其描述能力相当于下推自动机，语法形式如下：S->A，即S可以无条件的推导出A，与上下文无关。

正规文法也称为3型文法，等价于正则表达式，其描述能力相当于有穷自动机，语法形式如下：S->Aa，其中最后一个a必须为非终结符。

大多数程序语言的语法现象可用上下文无关文法描述。

在某C/C++程序中，整型变量a的值为0且应用在表达式“c=b/a”中，则最可能发生的情形是(50)。

(50)A.编译时报告有语法错误　　B.编译时报告有逻辑错误

C.运行时报告有语法错误　　D.运行时产生异常

【答案】D

【解析】本题考查程序语言基础知识。

对程序中含有变量的表达式求值发生在运行时，若除数为0进行除运算在运行时报告异常。

为了保证数据库中数据的安全可靠和正确有效，系统在进行事务处理时，对数据的插入、删除或修改的全部有关内容先写入(51)；当系统正常运行时，按一定的时间间隔，把数据库缓冲区内容写入(52)；当发生故障时，根据现场数据内容及相关文件来恢复系统的状态。

(51)A.索引文件　　B.数据文件　　C.日志文件　　D.数据字典

(52)A.索引文件　　B.数据文件　　C.日志文件　　D.数据字典

【答案】C　B

【解析】本题考查关系数据库事务处理方面的基础知识。

为了保证数据库中数据的安全可靠和正确有效，数据库管理系统（DBMS)提供数据库恢复、并发控制、数据完整性保护与数据安全性保护等功能。数据库在运行过程中由于软硬件故障可能造成数据被破坏，数据库恢复就是在尽可能短的时间内，把数据库恢复到故障发生前的状态。具体的实现方法有多种，如：定期将数据库作备份；在进行事务处理时，将数据更新（插入、删除、修改）的全部有关内容写入日志文件；当系统正常运行时，按一定的时间间隔，设立检查点文件，把内存缓冲区内容还未写入到磁盘中去的有关状态记录到检查点文件中；当发生故障时，根据现场数据内容、日志文件的故障前映像和检查点文件来恢复系统的状态。

“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”，这一性质通常被称为事务的(53)。

(53)A.原子性　　B.一致性　　C.隔离性　　D.持久性

【答案】C

【解析】本题考查数据库事务处理方面的知识。

事务具有原子性（atomicity)、一致性（consistency)、隔离性（isolation)和持久性(durability)。这 4 个特性也称事务的 ACID 性质。其中，事务的隔离性是指事务相互隔离，即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。

假定某企业 2014 年 5 月的员工工资如下表所示：

2014 年 5 月员工工资表

员工号	姓名	部门	基本工资	岗位工资	全勤奖	应发工资	扣款	实发工资
1001	王小龙	办公室	680.00	1200.00	100.00	1980.00	20.00	1960.00
1002	孙晓红	办公室	1200.00	1000.00	0.00	2200.00	50.00	2150.00
2001	赵晗珊	企划部	680.00	1200.00	100.00	1980.00	10.00	1970.00
2002	李丽敏	企划部	950.00	2000.00	100.00	3050.00	15.00	3035.00
3002	傅学君	设计部	800.00	1800.00	0.00	2600.00	50.00	2550.00
3003	曹海军	设计部	950.00	1600.00	100.00	2650.00	20.00	2630.00
3004	赵晓勇	设计部	1200.00	2500.00	0.00	3700.00	50.00	3650.00
4001	杨一凡	销售部	680.00	1000.00	100.00	1780.00	10.00	1770.00
4003	景昊星	销售部	1200.00	2200.00	100.00	3500.00	20.00	3480.00
4005	李建军	销售部	850.00	1800.00	100.00	2750.00	98.00	2652.00

查询人数大于 2 的部门和部门员工应发工资的平均工资的 SQL 语句如下：

```
SELECT ____(54)____
FROM   工资表
____(55)____
____(56)____;
```

(54)A.部门，AVG（应发工资）AS 平均工资　B.姓名，AVG（应发工资）AS 平均工资

C.部门，平均工资 AS AVG（应发工资）　D.姓名，平均工资 AS AVG（应发工资）

(55)A.ORDER BY 姓名　B.ORDER BY 部门　C.GROUP BY 姓名　D.GROUP BY 部门

(56)A.WHERE COUNT(姓名)>2　　B.WHERE COUNT(DISTINCT(部门)) >2

C.HAVING COUNT(姓名)>2　　D.HAVING COUNT(DISTINCT(部门)) >2

【答案】A　D

【解析】

(54)

本题考查 SQL 方面的基础知识。

查询各部门人数大于 2 和部门员工的平均工资的 SQL 语句如下：

```
SELECT 部门，AVG（应发工资）AS 平均工资
FROM  工资表
GROUP BY 部门
HAVING COUNT(姓名)> 2;
```

SQL 提供可为关系和属性重新命名的机制，这是通过使用具有“old-name as new-name”形式的 as 子句来实现的。As 子句即可出现在 select 子句，也可出现在 from 子句中。

(55) 因为本题是按部门进行分组，ORDER BY 子句的含义是对其后跟着的属性进行排序，故选项 A 和 B 均是错误的；GROUP BY 子句就是对元组进行分组，保留字 GROUP BY 后面跟着一个分组属性列表。根据题意，要查询部门员工的平均工资，选项 C 显然是错误的。

(56) 因为 WHERE 语句是对表进行条件限定，所以选项 A 和 B 均是错误的。在 GROUP BY 子句后面跟一个 HAVING 子句可以对元组在分组前按照某种方式加上限制。COUNT (*)是某个关系中所有元组数目之和，但 COUNT (A)却是 A 属性非空的元组个数之和。COUNT(DISTINCT(部门))的含义是对部门属性值相同的只统计 1 次。HAVING

COUNT(DISTINCT(部门))语句分类统计的结果均为1，故选项D是错误的；HAVING COUNT(姓名）语句是分类统计各部门员工，故正确的答案为选项C。

若对线性表的最常用操作是访问任意指定序号的元素，并在表尾加入和删除元素，则适宜采用(57)存储。

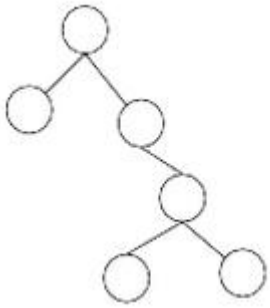
(57)A.顺序表　　B.单链表　　C.双向链表　　D.哈希表

【答案】A

【解析】本题考查数据结构基础知识。

线性表的元素在逻辑上是一个线性序列，若最常用的操作是访问任意指定序号的元素，而且其插入和删除元素的操作均在表尾进行，不需要移动其他元素，则其存储结构采用顺序表最为合适。

某二叉树如图所示，若进行顺序存储（即用一维数组元素存储该二叉树中的结点且通过下标反映结点间的关系，例如，对于下标为i的结点，其左孩子的下标为2i、右孩子的下标为2i+1），则该数组的大小至少为（58）；若采用三叉链表存储该二叉树（各个结点包括结点的数据、父结点指针、左孩子指针、右孩子指针），则该链表的所有结点中空指针的数目为（59）。



(58)A.6　　B.10　　C.12　　D.15

(59)A.6　　B.8　　C.12　　D.14

【答案】D　B

【解析】本题考查数据结构基础知识。

题图所示的二叉树有6个结点，根结点的编号为1，其左孩子和右孩子分别为2和3,按照右孩子链继续，3号结点的右孩子编号为7,7号结点的右孩子编号为15,因此该二叉树进行顺序存储时数组大小至少为15。采用三叉链表存储时，每个结点有3个指针域，共18

个指针域，其中，12 个孩子指针用了 5 个，剩余 7 个为空指针，6 个父结点指针用了 5 个，剩余 1 个为空（即根结点无双亲），因此，结点中的指针域有 8 个为空。

某双端队列如下图所示，要求元素进出队列必须在同一端口，即从 A 端进入的元素必须从 A 端出、从 B 端进入的元素必须从 B 端出，则对于 4 个元素的序列 e1、 e2、e3、 e4，若要求前 2 个元素(e1、 e2)从 A 端口按次序全部进入队列，后两个元素(e3、e4)从 B 端口按次序全部进入队列，则可能得到的出队序列是（60）。

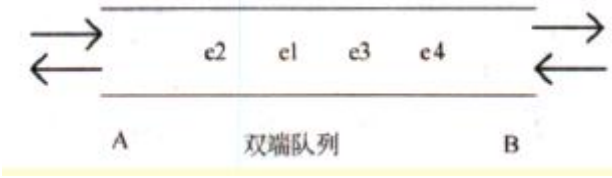
(60)A.e1、 e2、 e3、 e4　　B.e2、 e3、 e4、 e1

C.e3、 e4、e1、 e2　　D.e4、 e3、 e2、 e1

【答案】D

【解析】本题考查数据结构基础知识。

按照题目所述，当 e1、e2 从 A 端口按次序全部进入队列，e3、e4 从 B 端口按次序全部进入队列后，双端队列的状态如下图所示。



在这种情形下，e1 和 e3 不可能先出队列，所以排除选项 A 和 C。若 e2 先出队列，则剩下的 3 个元素中，只能是 e1 或 e4 出队列，所以 e2、e3、e4、e1 是不可能的出队序列，这样就排除了选项 B。选项 D 的 e4、e3、e2、e1 是可能的出队序列。

实现二分查找（折半查找）时，要求查找表(61)。

(61)A.顺序存储，关键码无序排列　　B.顺序存储，关键码有序排列

C.双向链表存储，关键码无序排列　　D.双向链表存储，关键码有序排列

【答案】B

【解析】本题考查数据结构基础知识。

二分查找是一种高效的查找方法，其思路是待查找元素先与序列中间位置上的元素比较，

若相等，则查找成功：若待查找元素较大，则接下来到序列的后半区进行二分查找，否则到序列的前半区进行二分查找。显然，要快速定位序列的中间位置，查找表必须进行顺序存储；其次，从二分查找过程可知，序列必须有序排列才行。

某个算法的时间复杂度递归式 T(n)=T(n-1)+n，其中 n 为问题的规模，则该算法的渐进时间复杂度为（62），若问题的规模增加了 16 倍，则运行时间增加（63）倍。

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

(63)A.16　　B.64　　C.256　　D.1024

【答案】C　C

【解析】本题考查算法分析的基础知识。

直接展开递归式 T(n)=T(n-1)+n

$$=T(n-2)+(n-1)+n$$

$$=T(n-3)+(n-2)+(n-1)+n$$

$$=1+2+\ldots+n$$

$$=n(n-1)/2$$

$$=\Theta(n^2)$$

得到该算法的时间复杂度为 $\Theta(n^2)$，当问题的规模增加了 16 倍时，运行时间增加了 162=256 倍。

Prim 算法和 Kruscal 算法都是无向连通网的最小生成树的算法，Prim 算法从一个顶点开始，每次从剩余的顶点中加入一个顶点，该顶点与当前的生成树中的顶点的连边权重最小，直到得到一颗最小生成树；Kruscal 算法从权重最小的边开始，每次从不在当前的生成树顶点中选择权重最小的边加入，直到得到一颗最小生成树，这两个算法都采用了(64)设计策略，且（65）。

(64)A.分治　　B.贪心　　C.动态规划　　D.回溯

(65)A.若网较稠密，则 Prim 算法更好　　B.两个算法得到的最小生成树是一样的

C.Prim 算法比 Kruscal 算法效率更高　　D.Kruscal 算法比 Prim 算法效率更高

【答案】B　A

【解析】本题考查算法设计与分析的基础知识。

Prim 算法从扩展顶点开始，每次总是“贪心的”选择与当前顶点集合中距离域短的顶点，而 Kruscal 算法从扩展边开始，每次总是“贪心的”选择剩余的边中最小权重的边，因此两个算法都是基于贪心策略进行的。

Prim 算法的时间复杂度为 O(n2)，其中 n 为图的顶点数，该算法的计算时间与图中的边数无关，因此该算法适合于求边稠密的图的最小生成树：Kruscal 算法的时间复杂度为 O(mlgm)，其中 m 为图的边数，该算法的计算时间与图中的顶点数无关，因此该算法适合于求边稀疏的图的最小生成树。当图稠密时，用 Prim 算法效率更高。但若事先没有关于图的拓扑特征信息时，无法判断两者的优劣。由于一个图的最小生成树可能有多棵，因此不能保证用这两种算法得到的是同一棵最小生成树。

IP 地址块 155.32.80.192/26 包含了(66)个主机地址，以下 IP 地址中，不属于这个网络的地址是(67)。

(66)A.15　　B.32　　C.62　　D.64

(67)A.155.32.80.202　　B.155.32.80.195

C.155.32.80.253　　D.155.32.80.191

【答案】C　D

【解析】

地址块 155.32.80.192/26 包含了 6 位主机地址，所以包含的主机地址为 62 个。

网络地址 155.32.80.192/26 的二进制为: 10011011 00100000 01010000 11000000

地址 155.32.80.202 的二进制为：10011011 00100000 01010000 11001010

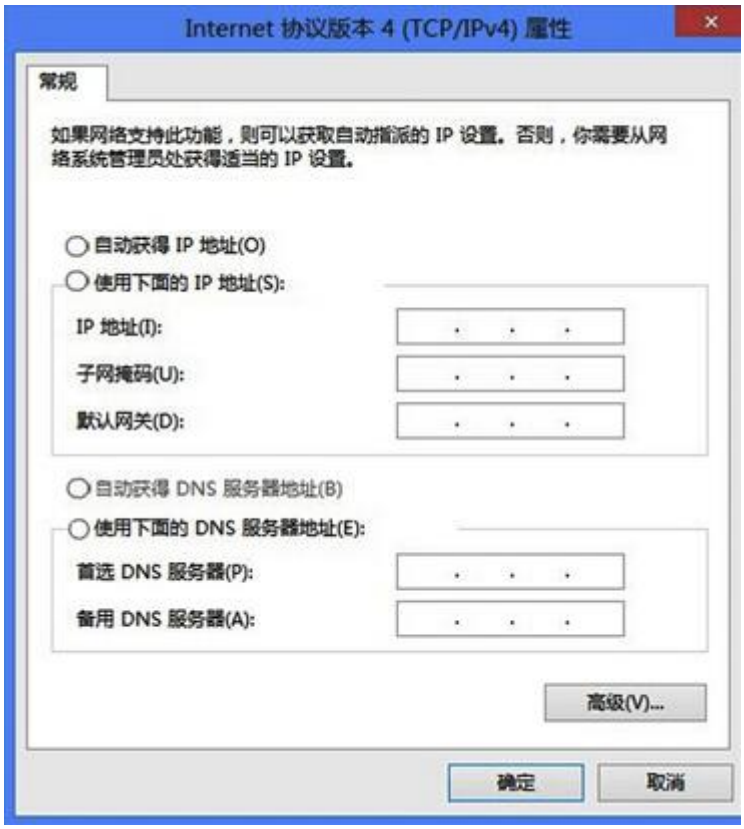
地址 155.32.80.191 的二进制为：10011011 00100000 01010000 10111111

地址 155.32.80.253 的二进制为：10011011 00100000 01010000 11111101

地址 155.32.80.195 的二进制为：10011011 00100000 01010000 11000011

可以看出，地址 155.32.80.191 不属于网络 155.32.80.192/26。

校园网连接运营商的 IP 地址为 202.117.113.3/30，本地网关的地址为 192.168.1.254/24，如果本地计算机采用动态地址分配，在下图中应该如何配置？(68)。



(68)A.选取“自动获得 IP 地址”

B.配置本地计算机 IP 地址为 192.168.1.×

C.配置本地计算机 IP 地址为 202.115.113.×

D.在网络 169.254.×.×中选取一个不冲突的 IP 地址

【答案】A

【解析】

如果采用动态地址分配方案，本地计算机应设置为“自动获得 IP 地址”。

某用户在使用校园网中的一台计算机访问某网站时，发现使用域名不能访问该网站，但是使用该网站的 IP 地址可以访问该网站，造成该故障产生的原因有很多，其中不包括(69)。

(69)A.该计算机设置的本地 DNS 服务器工作不正常

B.该计算机的 DNS 服务器设置错误

C.该计算机与 DNS 服务器不在同一子网

D.本地 DNS 服务器网络连接中断

【答案】C

【解析】本题主要考查网络故障判断的相关知识。

如果本地的DNS服务器工作不正常或者本地DNS服务器网络连接中断都有可能导致该计算机的DNS无法解析域名，而如果直接将该计算机的DNS服务器设置错误也会导致DNS无法解析域名，从而出现使用域名不能访问该网站，但是使用该网站的IP地址可以访问该网站。但是该计算机与DNS服务器不在同一子网不会导致DNS无法解析域名的现象发生，通常情况下大型网络里面的上网计算机与DNS服务器本身就不在一个子网，只要路由可达DNS都可以正常工作。

中国自主研发的3G通信标准是(70)。

(70)A.CDMA2000　　B.TD-SCDMA　　`C.WCDMA　　D.WiMAX

【答案】B

【解析】

1985年，ITU提出了对第三代移动通信标准的需求，1996年正式命名为IMT-2000(International Mobile Telecommunications-2000),其中的2000有3层含义：

使用的频段在2000MHz附近。

通信速率约为2000kb/s (即2Mb/s)。

预期在2000年推广商用。

1999年ITU批准了五个IMT-2000的无线电接口，这五个标准是：

IMT-DS(Direct Spread)：即W-CDMA,属于频分双工模式，在日本和欧洲制定的UMTS系统中使用。

IMT-MC(Multi-Carrier)：即CDMA-2000,属于频分双工模式，是第二代CDMA系统的继承者。

IMT-TC(Time-Codc)：这一标准是中国提出的TD-SCDMA，属于时分双工模式。

IMT-SC(Single Carrier)：也称为EDGE,是一种2.75G技术。

IMT-FT(Frequency Time)：也称为DECT。

2007年10月19日，ITU会议批准移动WiMAX作为第6个3G标准，称为IMT-2000 OFDMATDDWMAN，即无线城域网技术。

第三代数字蜂窝通信系统提供第二代蜂窝通信系统提供的所有业务类型，并支持移动多媒体业务。在高速车辆行驶时支持144kb/s的数据速率，步行和慢速移动环境下支持384kb/s的数据速率，室内静止环境下支持2Mb/s的高速数据传输，并保证可靠的服务质量。

Cloud computing is a phrase used to describe a variety of computing concepts that involve a large number of computers (71) through a real-time commutinication network such as the Internet. In science, cloud computing is a (72) for distributed computing over a network, and means the (73) to run a program or application on many connected computers at the same time. The architecture of a cloud is developed at three layers: infrastructure, platform, and application.The infrastucture layer is built with virtualized compute,storage,and network resources. The platform layer is for general-purpose and repeated usage of the collection of software resources. The application layer is formed with a collection of all needed software modules for SaaS applications. The infrastucture layer serves as the (74) for building the platform layer of the cloud. In turn,the platform layer is foundation for implementing the (75) layer for SaaS application.

(71)A.connected	B.implemented	C.optimized	D.virtualized
(72)A.replacement	B.switch	C.substitute	D.synonym
(73)A.ability	B.approach	C.function	D.method
(74)A.network	B.foundation	C.software	D.hardware
(75)A.resource	B.service	C.application	D.software

【答案】A D A B C

【解析】本题考查对英语资料的阅读理解。

本段英文简要介绍云计算相关概念。

云计算是用来描述各种计算概念的短语，包括大量计算机通过网络相互连接以实现分布计算，意思是同时在很多互联的计算机上运行程序或应用的能力。

云的架构分为基础设施层、平台层和应用层三层。基础设施层由虚拟计算、存储和网络资源构成。平台层用于一组软件资源重复使用的通用目的。应用层由一组所需的软件模块构成，即软件即服务（SaaS)。基础设施层作为构建平台层的基础。相反，平台层是应用层的基础，为SaaS应用实现应用层。

试题一（共 15 分）

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】

某巴士维修连锁公司欲开发巴士维修系统，以维护与维修相关的信息。该系统的主要功能如下：

1)记录巴士 ID 和维修问题。巴士到车库进行维修，系统将巴士基本信息和 ID 记录在巴士列表文件中，将待维修机械问题记录在维修记录文件中，并生成维修订单。

2)确定所需部件。根据维修订单确定维修所需部件，并在部件清单中进行标记。

3)完成维修。机械师根据维修记录文件中的待维修机械问题，完成对巴士的维修，登记维修情况；将机械问题维修情况记录在维修记录文件中，将所用部件记录在部件清单中，并将所用部件清单发送给库存管理系统以对部件使用情况进行监控。巴士司机可查看已维修机械问题。

4)记录维修工时。将机械师提供的维修工时记录在人事档案中；将维修总结发送给主管进行绩效考核。

5)计算维修总成本。计算部件清单中实际所用部件、人事档案中所用维修工时的总成本；将维修工时和所用部件成本详细信息给会计进行计费。 现采用结构化方法对巴士维修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。

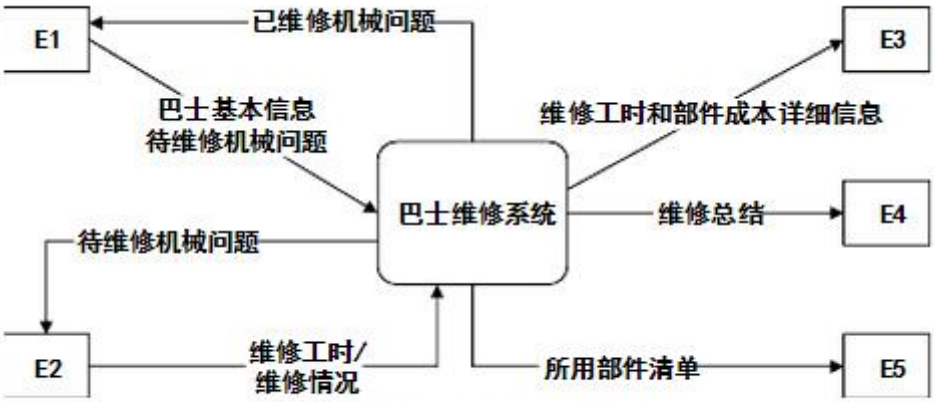


图 1-1 上下文数据流图

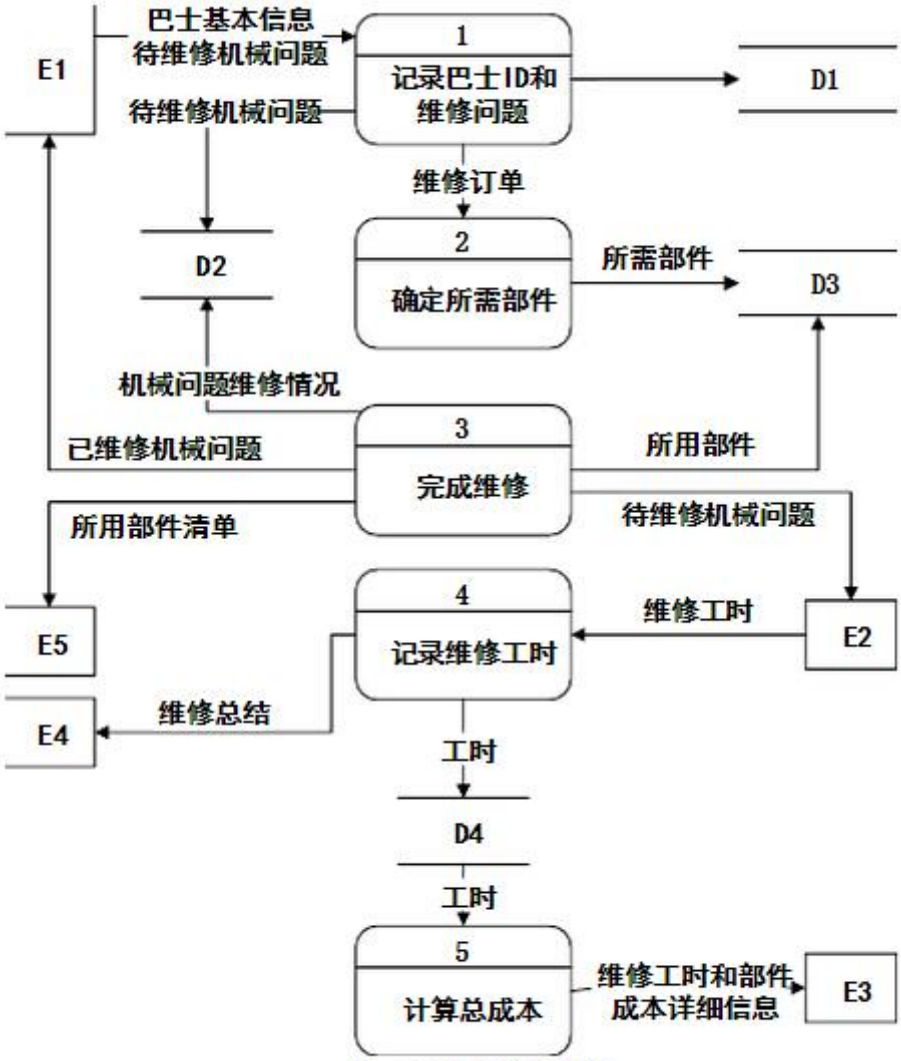


图 1-2 0 层数据流图

【问题 1】

使用说明中的词语，给出图 1-1 中的实体 E1～E5 的名称。

E1：巴士司机

E2：机械师

E3：会计

E4：主管

E5：库存管理系数

【试题解析】

本题考查的是 DFD 的应用，属于比较传统的题目，考查点也与往年类似。

本问题考查的是顶层 DFD。顶层 DFD 通常用来确定系统边界，其中只包含一个唯一的加工（即

待开发的系统)、外部实体以及外部实体与系统之间的输入输出数据流。题目要求填充的正是外部实体。

从题干说明 1）没有明确说明由巴士到车库后由谁提供待维修问题，图 1-1 中的 E1，考察说明中 3)中最后一句说明“巴士司机可查看已维修机械问题”可以看出，从系统到巴士司机有输出数据流“已维修机械问题”，可知 E1 为巴士司机。从 2)中“机械师根据维修记录文件中的待维修机械问题，完成对巴士的维修，登记维修情况”；再看说明 4)中机械师提供维修工时，可以看出，从 E2 到系统有输入数据流“维修工时”、输出数据流“待维修机械问题”，可知 E2 为机械师，还将维修总结发送给主管，即系统到 E4 有输出数据流“维系总结”，可知 E4 为主管。从说明 5)将维修工时和所用部件成本详细信息给会计，从系统到 E3 有输出数据流“维修工时和所用部件成本详细信息”，可知 E3 为会计。说明 3)中将所用部件清单发送给库存管理系统以对部件使用情况进行监控，及系统到 E5 有输出数据流“所用部件清单”，可知 E5 为库存管理系统。

【问题 2】

使用说明中的词语，给出图 1-2 中的数据存储 D1～D4 的名称。

D1：巴士列表文件

D2：维修记录文件

D3：部件清单

D4：人事档案

【试题解析】

本问题考查 0 层数据流图中的数据存储。系统中的主要功能与图 1-2 中的处理一一对应，1）对应处理“ 记录巴士 ID 和维修问题”，将巴士 ID 记录在巴士列表文件中，可知 D1 为巴士列表文件。说明 2）对应处理 “ 确定所需部件 ”，将维修所需部件在部件清单中进行标记，可知以 D3 为部件清单。说明 1）中将待维修机械问题记录在维修记录文件中，可知 D2 为维修记录文件。说明 4）对应处理“ 记录维修工时 “，描述了将机械师提供的维修工时记录在人事档案中，可以判定 D4 是人事档案。

【问题 3】

说明图 1-2 中所存在的问题。

图 1-2 中处理 3 只有输出数据流，没有输入数据流。D2 和 D3 是黑洞，只有输入的数据流，没有输出的数据流。父图与子图不平衡，图 1-2 中没有图 1-1 中的数据流“维 修情况”。

【试题分析】

本问题考查 0 层数据流图中的数据流。分析图 1-2,可以发现，处理 3 只有输出数据流没有输入数据流，D2 和 D3 只有输入数据流，而没有输出流，造成黑洞。另外，对 照图 1-2 和图 1-1，发现图 1-1 中从 E2 输入的数据流维修工时/维修情况，在图 1-2 中只有维修工时，造成父图与子图不平衡。

【问题 4】

根据说明和图中术语，采用补充数据流的方式，改正图 1-2 中的问题。要求给出所补充数据流的名称、起点和终点。

数据流名称	起　点	终　点
待维修机械问题	D2 或维修记录文件	3 或完成维修
实际所用部件	D3 或部件清单	5 或计算总成本
维修情况	E2 或机械师	3 或完成维修

【试题分析】

针对【问题 3】分析图 1-2 中存在的问题，题目要求以补充数据流的方式解决，进一步分析说明，说明 3)对应处理“完成维修”，机械师根据维修记录文件中的待维修机械问题完成对巴士的维修，可知处理完成维修需要从维修记录文件读取待维修问题，补充一条从 D2 到处理 3 的数据流“待维修机械问题”。说明 5)对应处理“计算维修总成本”，需要计算部件清单中实际所用部件，补充从部件清单到计算总成本的数据流“实际所用部件”。说明 3)中机械师要登记维修情况，判定图 1-2 中缺少了 E2 到处理 3 的数据流“维修情况”。

到此为止所有缺失的数据流都补齐了，也解决了【问题 3】中的平衡问题、处理只有输出数据流没有输入数据流的问题，D2 和 D3 也既有输入数据流，又有输出数据流。

试题二（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3,将解答填入答题纸的对应栏内。

【说明】

某家电销售电子商务公司拟开发一套信息管理系统，以方便对公司的员工、家电销售、家电厂商和客户等进行管理。

【需求分析】

(1)系统需要维护电子商务公司的员工信息、客户信息、家电信息和家电厂商信息等。员工信息主要包括：工号、姓名、性别、岗位、身份证号、电话、住址，其中岗位包括部门经理和客服等。客户信息主要包括：客户 ID、姓名、身份证号、电话、住址、账户余额。家电信息主要包括：家电条码、家电名称、价格、出厂日期、所属厂商。家电厂商信息包括：厂商 ID、厂商名称、电话、法人代表信息、厂址。

(2)电子商务公司根据销售情况，由部门经理向家电厂商订购各类家电。每个家电厂商只能由一名部门经理负责。

(3)客户通过浏览电子商务公司网站查询家电信息，与客服沟通获得优惠后，在线购买。

【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。

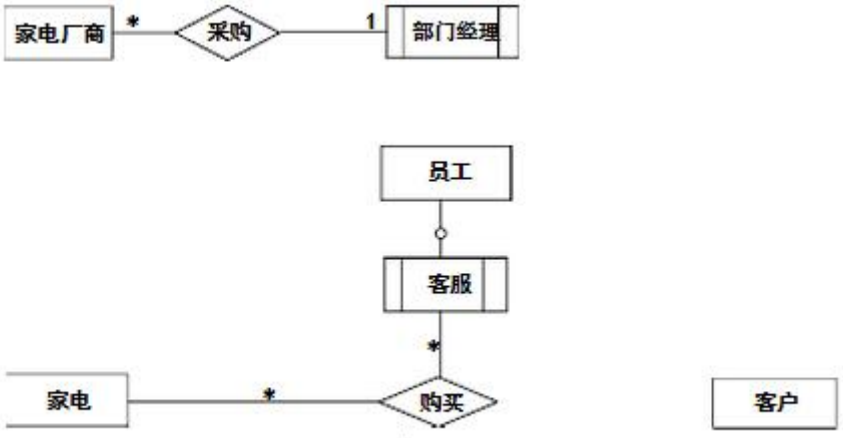


图 2-1 实体联系图

【逻辑结构设计】

根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）：

客户(客户 ID、姓名、身份证号、电话、住址、账户余额)

员工（工号、姓名、性别、岗位、身份证号、电话、住址）

家电(家电条码、家电名称、价格、出厂日期、 (1))

家电厂商(厂商 ID、厂商名称、电话、法人代表信息、厂址、 (2))

购买(订购单号、 (3) 、金额)

【问题 1】

补充图 2-1 中的联系和联系的类型。

【试题分析】

本题考查数据库设计，属于比较传统的题目，考查点也与往年类似。

本问题考查数据库的概念结构设计，题目要求补充完整实体联系图中的联系和联系的类型。

根据题目的需求描述可知，一个家电厂商可以供应多台家电，而一台家电只能对应一个家电厂商，因此“家电厂商”和“家电”之间存在“供应”联系，联系的类型为一对多（1:*，或 1:m)。

根据题目的需求描述可知，“员工”和“部门经理”之间存在一个包含关系。

根据题目的需求描述可知，“客户”、“客服”和“家电’’之间存在“购买”联系，联系的类型为多对多对多（*:*:*，或 m:n:o)。

【问题 2】

根据图 2-1，将逻辑结构设计阶段生成的关系模式中的空(1)～(3)补充完整。用下划线指出“家电”、“家电厂商”和“购买”关系模式的主键。

【参考答案】

（1）厂商 ID

（2）部门经理工号 或 经理工号 或 员工工号

（3）家电条码，客户 ID，客服工号

关系模式	主键
家电	家电条码
家电厂商	厂商 ID
购买	订购单号

【试题分析】

本问题考查数据库的逻辑结构设计，题目要求补充完整各关系模式，并给出各关系模式的主键。

根据实体联系图和需求描述，“家电”和“家电厂商”存在多对一的关系，在家电关系中尚要记录家电厂商的主键，也就是“厂商 ID”。所以，对于“家电”关系模式，需补充属性“厂商 ID”。“家电条码”为“家电”关系的主键。

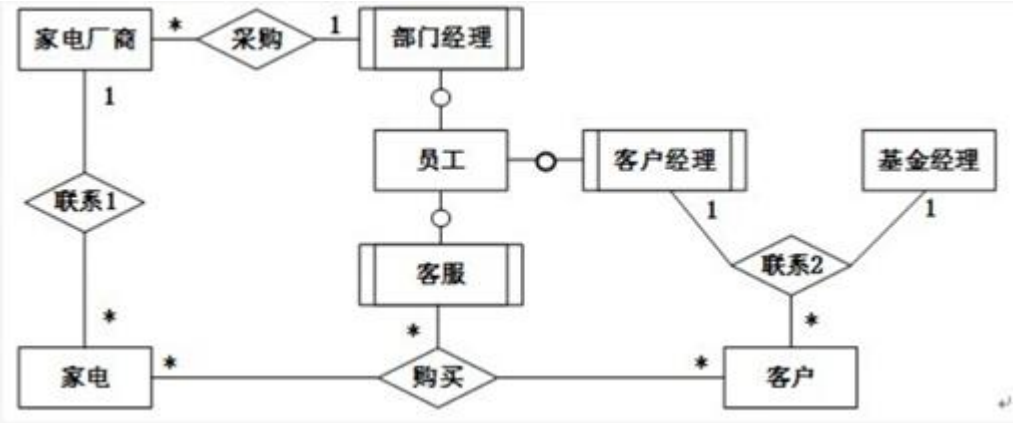
根据实体联系图和需求描述，“家电厂商”和“部门经理”之间存在多对一的关系，在家电厂商关系中需要记录部门经理的主键，也就是“部门经理工号”（或“经理工号”、或“员工工号”）。“厂商 ID”为“家电厂商”的主键。

根据实体联系图和需求描述，“客户”、“客服”和“家电”之间的多对多对多的“购买”联系。因为是多对多对多联系，所以“购买”联系需要单独作为一个关系，这个关系需要 id 录“客户”、“客服”和“家电”的主键。所以，对于“购买”关系模式，需补充属性“客户 ID” “客服工号”和“家电条码”。“订购单号”为“购买”的主键。

【问题 3】

电子商务公司的主营业务是销售各类家电，对账户有余额的客户，还可以联合第三方基金公司提供理财服务，为此设立客户经理岗位。客户通过电子商务公司的客户经理和基金公司的基金经理进行理财。每名客户只有一名客户经理和一名基金经理负责，客户经理和基金经理均可负责多名客户。请根据该要求，对图 2-1 进行修改，画出修改后的实体间联系和联系的类型。

【参考答案】



【试题分析】

本问题考查数据库的概念结构设计，根据新增的需求增加实体联系图中的实体的联系和联系的类型。

根据问题描述，需要新增“客户经理”，包含于“员工”。

根据问题描述，客户只由一名客户经理和一名基金经理负责，客户经理和基金经理均可负责多名客户，所以“客户”、“客户经理”和“基金经理”之间存在一个“理财”联系，联系的类型为多对 1 对 1（*:1:1， 或 m:1:1）。

试题三（共 15 分）

阅读下列说明和图，回答问题 1 至问题 3,将解答填入答题纸的对应栏内。

【说明】

某高校图书馆欲建设一个图书馆管理系统，目前已经完成了需求分析阶段的工作。功能需求均使用用例进行描述，其中用例“借书(Check Out Books)”的详细描述如下。

参与者：读者(Patron)。

典型事件流：

1．输入读者 ID；

2．确认该读者能够借阅图书，并记录读者 ID；

3．输入所要借阅的图书 ID；

4．根据图书目录中的图书 ID 确认该书可以借阅，计算归还时间，生成借阅记录；

5．通知读者图书归还时间。

重复步骤 3-5，直到读者结束借阅图书。

备选事件流：

2a.若读者不能借阅图书，说明读者违反了图书馆的借书制度（例如，没有支付借书费用等）

①告知读者不能借阅，并说明拒绝借阅的原因；

②本用例结束。

4a.读者要借阅的书无法外借

①告知读者本书无法借阅；

②回到步骤 3。

说明：图书的归还时间与读者的身份有关。如果读者是教师，图书可以借阅一年；如果是学生，则只能借阅 3 个月。读者 ID 中包含读者身份信息。

现采用面向对象方法开发该系统，得到如图 3-1 所示的系统类模型（部分）；以及如图 3-2 所示的系统操作“checkOut(bookID)（借书）”的通信图（或协作图）。

【问题 1】

根据说明中的描述，以及图 3-1 和图 3-2，给出图 3-1 中 C1～C4 处所对应的类名（类名使用图 3-1 和图 3-2 中给出的英文词汇）。

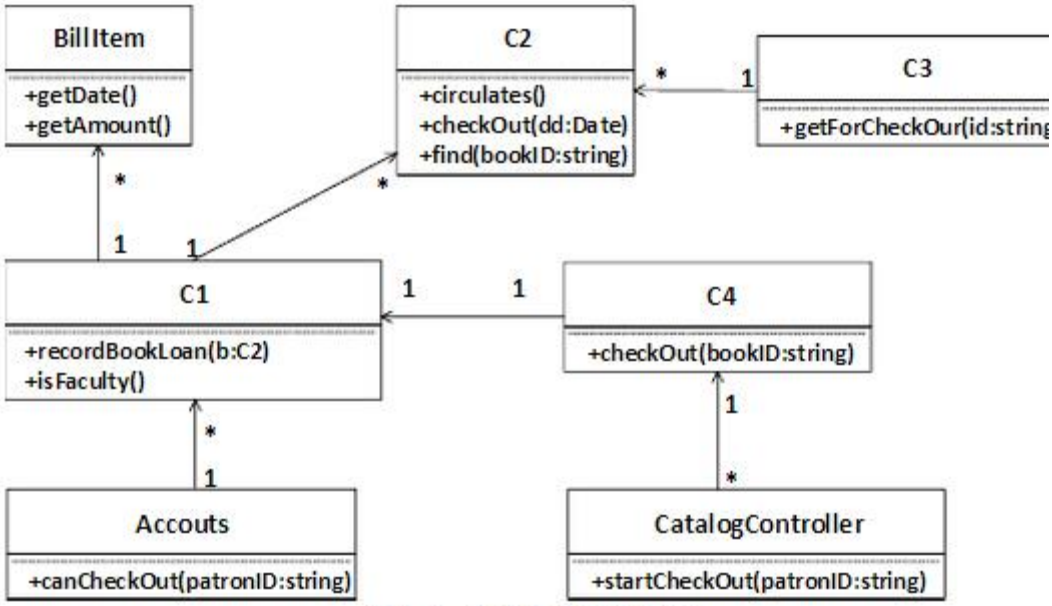


图 3-1 系统类模型（部分）

图 3-1 系统类模型（部分）

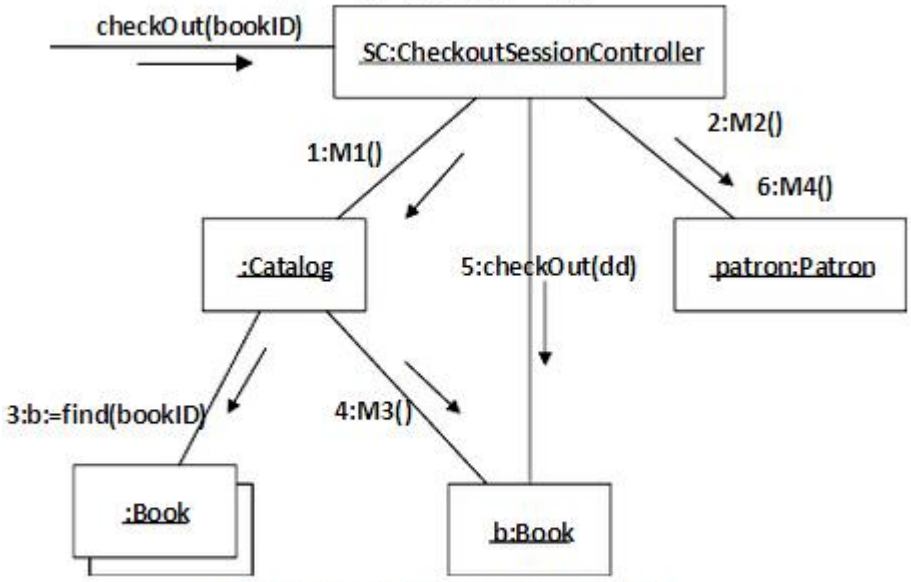


图 3-2 系统操作 checkOut 的通信图

【参考答案】

C1：Patron

C2：Book

C3：Catalog

C4：Check out Session controller

【试题分析】

本题属于经典的考题，主要考查面向对象分析方法以及 UML 类图和通信图的相关知识。

说明中给出了一个具体用例的详细描述，给出了其中的一个系统操作“checkout bookID)(借书)”的通信图，需要考生利用通信图中的信息来补充类图中缺失的部分。 通信图（communication diagram)强调收发消息的对象的结构组织，在早期的版本中也被称作协作图。通信图强调参加交互的对象的组织。产生一张通信图，首先要将参加交互的对象作为图的顶点，然后把连接这些对象的链表示为阁的弧，最后用对象发送和接收的消息来修饰这些链。这就提供了在协作对象的结构组织的语境中观察控制流的一个清晰的可视化轨迹。

消息 checkOut(booklD)的接收者是类 CheckoutSessionContrailer 的对象，说明类 CheckoutSessionController 中应该包含这一方法，否则无法响应该条消息。由图 3-1 可知，C4 处所代表的类应该是 CheckoutSessionController。

消息 find(booklD)的接收者是类 Book，同理，由图 3-1 可知，C2 处对应的类应该是 Book。根据用例描述，图书信息是包含在图书目录中，所以 C3 处对应的类应该是 Catalog，C1 处对应的就应该是 Patron 了。

【问题 2】

根据说明中的描述，以及图 3-1 和图 3-2，给出图 3-2 中 M1～M4 处所对应的方法名（方法名使用图 3-1 和图 3-2 中给出的英文词汇）。

【参考答案】

M1:getforcheckout

M2:isFaculty

M3:circulates

M4:recordBookLoan

【试题分析】

图 3-1 填充完整之后，图 3-2 的空缺就比较容易填写了。在通信图中，对象之间传递的消息就对应着接收对象中的方法。M1 对应的就是类 Catalog 中的方法，由图 3-1 可 知，M1 对应的是 getForCheckOut。

M3 对应的应该是类 Book 中的方法。由图 3-1 可知，Book 中有 3 个方法，find 和 checkout 已经出现在通信图上了，所以 M3 应该是 circulates。

M2 和 M4 是类 Patron 中的方法。Patron 中有 2 个方法。通信图中的消息是有序号的，这个序号表示了消息的时间顺序，也就是说发送 M2 的时间要早于消息 M4，因此必须区分类 Patron 中两个方法使用的先后顺序。在用例描述中特别指出：图书的归还时间与读者的身份有关。计算还书及借书费用时，需先确定读者的身份，因此方法 isFaculty 应该先被调用，所以 M2 对应 isFaculty，M4 对应 recordBookLoan。

【问题 3】

用例“借书”的备选事件流 4a 中，根据借书制度来判定读者能否借阅图书。若图书馆的借书制度会不断地扩充，并需要根据图书馆的实际运行情况来调整具体使用哪些制度。为满足这一要求，在原有类设计的基础上，可以采用何种设计模式？简要说明原因。

【参考答案】

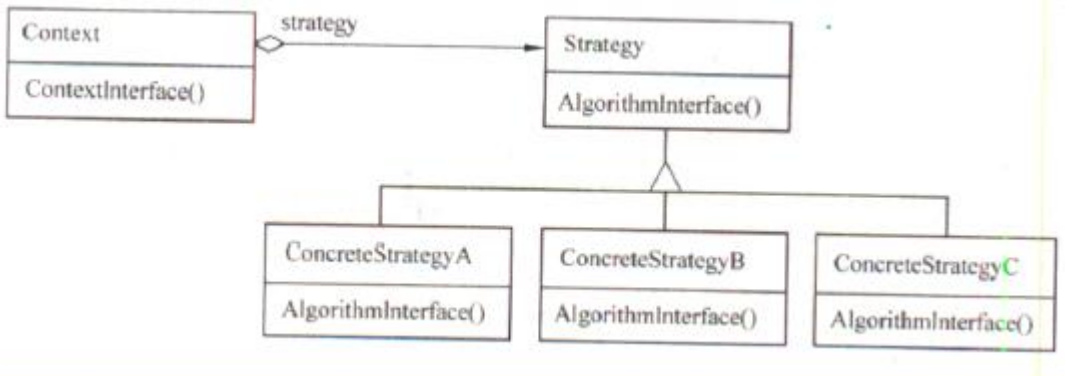
策略模式

策略模式定义了一系列算法，并将每个算法封装起来，而且使它们可以相互替换。 策略模式让算法独立于使用它们的客户而变化。适用于需要在不同情况下使用不同的策略（算法），或者策略还可能在未来用其他方式来实现。

【试题解析】

本题在设计类时使用到了策略模式。

策略模式定义了一系列的算法，把它们一个个封装起来，并且使它们可以相互替换。此模式使得算法可以独立于使用它们的客户而变化。策略模式的结构如下图所示。



其中：

- Strategy（策略）定义所有支持的算法的公共接口。Context 使用这个接口来调用某

ConcreteStrategy 定义的算法。

• ConcreteStrategy（具体策略）以 Strategy 接口实现某具体算法。

• Context（上下文）用一个 ConcreteStrategy 对象来配置；维护一个对 Strategy 对象的引用；可定义一个接口来让 Strategy 访问它的数据。

Strategy 模式适用于：

•许多相关的类仅仅是行为有异。“策略”提供了一种用多个行为中的一个行为来配置一个类的方法。

•需要使用一个算法的不同变体。例如，定义一些反应不同空间的空间/时间权衡的算法。当这些变体实现为一个算法的类层次时，可以使用策略模式。

•算法使用客户不应该知道的数据。可使用策略模式以避免暴露复杂的、与算法相关的数据结构。

• 一个类定义了多种行为，并且这些行为在这个类的操作中以多个条件语句的形式出现，将相关的条件分支移入它们各自的 Strategy 类中，以代替这些条件语句。

试题四（共 15 分）

阅读下列说明和 C 代码，回答问题 1 至问题 3,将解答写在答题纸的对应栏内。

【说明】

采用归并排序对 n 个元素进行递增排序时，首先将 n 个元素的数组分成各含 n/2 个元素的两个子数组，然后用归并排序对两个子数组进行递归排序，最后合并两个已经排好序的子数组得到排序结果。 下面的 C 代码是对上述归并算法的实现，其中的常量和变量说明如下：

arr：待排序数组

p，q，r：一个子数组的位置为从 p 到 q，另一个子数组的位置为从 q+1 到 r

begin，end:待排序数组的起止位置

left，right:临时存放待合并的两个子数组

n1，n2:两个子数组的长度

i，j，k：循环变量

mid:临时变量

【C 代码】

```
#include<stdio.h>
#include<stdlib.h>
#define MAX 65536
void merge(int arr[],int p,int q,int r) {
     int *left,*right;
     int n1,n2,i,j,k;
     n1=q-p+1;
     n2=r-q;
     if((left=(int*)malloc((n1+1)*sizeof(int)))==NULL){
        perror("malloc error");
        exit(1);
      }
     if((right=(int*)malloc((n2+1)*sizeof(int)))==NULL){
        perror("malloc error");
         exit(1);
      }
      for(i=0;i<n1;i++){
         left[i] = arr[p+i];
       }
      left[i] = MAX;
      for(i=0;i<n2;i++){
        right[i] = arr[q+i+1];
      }
```

```
        right[i]= MAX;
        i=0; j=0;
        for (k=p;  (1)  k++ ) {
         if(left[i]> right[j]{
            (2)
          j++;
         }else{
          arr[k]= left[i];
          i++;
        }
    }
}
void mergeSort(int arr[],int begin,int end){
    int mid;
    if (  (3)  ){
    mid=(begin+ end)/2;
    mergeSort(arr,begin,mid);
        (4)
    merge(arr,begin,mid,end);
    }
}
```

【问题 1】

根据以上说明和 C 代码，填充 C 代码中的空(1)～(4)。

【参考答案】

1）k≤r 或 k<r+1

2）arr[k]=right[j]

3）begin<end

4)mergeSort(arr,mid+1,end)

【试题分析】

本题考查算法设计、分析和 C 程序实现的知识，属于传统题目，考查点也与往年类似。

归并排序是一种经典的排序算法，基本思想：把 n 个元素构成的数组分成两个 n/2 个元素构成的子数组，再进一步划分，一直到每个子数组仅包含 1 个元素，此时再把两 两有序的数组合并成更大的有序数组，一直到整个数组有序为止。

本问题考查算法的实现。C 程序中有两个函数，merge 函数将两个有序数组合并成一个更大的有序数组。归并过程是首先将两个有序的子数组的元素分别放到 left 和 right 数组中，然后依次比较这两个数组中的元素，从小到大把元素放到 arr 数组的特定元素中。包含空格

(1)的 for 循环中，给出了将 left 和 right 元素放入 arr 中，放入的位置是从 p 到 r，因此，空格（1)填写 k <= r。在比较 left[i]和 right[j]元素时，若 left[i] > right[j]， 则应该把 right[j]的值放入 arr[k]中，因此空格（2)填写 arr[k] = right[j]。mergeSort 函数进行数组的排序。若数组元素个数大于 1，则继续划分，因此空格（3)填写 begin < end。将数组从 arr[begin]到 arr[end]划分为 anfbegin]到 arr[mid]和 arr[mid + 1]到 arr[end]两个部分，因此空格(4)填写 mergeSort(arr，mid + 1，end)。

【问题 2】

根据题干说明和以上 c 代码，算法采用了 (5) 算法设计策略。

分析时间复杂度时，列出其递归式为 (6) ，解得渐进时间复杂度为 (7) （用 0 符号表示)。空间复杂度为(8)（用 0 符号表示)。

【参考答案】

5) 分治

6) T(n)=2T(N/2)+0(n)或 T(n)=2T(n/2)+f(n) （ f(n)为线性函数）

7) 0(nlogn)

8) 0(n)

【试题分析】

本问题考查算法的设计策略和时间复杂度，归并排序算法是一个典型的分治算法。每次将一个规模为 n 的问题变成两个规模为 n/2 的子问题，划分时直接从中间分开，因此划分采用 0(1)的时间，然后是递归求解两个子问题，根据 merge 函数代码，合并的时间是线性时间 0(n)。因此递归式为 T(n)=2T(n/2)+f(n)，f(n)为线性函数。用主方法求解，得到时间复杂度为 0(nlgn)。由于在归并过程中，需要 left 和 right 两个辅助数组，其规模为待排序的数组长度，即 0(n)。

【问题 3】

两个长度分别为 n1 和 n2 的已排好序的子数组进行归并，根据上述 C 代码，则元素之间比较次数为 (9) 。

【参考答案】

(9) n1+n2

【试题分析】

本问题考查对算法的进一步分析。元素之间的比较次数就是 merge 函数的最后一个 for 循环体执行的次数，由于 k 从 p 到 r，故循环体执行次数为 r-p+1 次，即 n1+n2 次。

试题五（共 15 分）

【说明】

某实验室欲建立一个实验室环境监测系统，能够显示实验室的温度、湿度以及洁净度等环境数据。当获取到最新的环境测量数据时，显示的环境数据能够更新。 现在采用观察者(Observer)模式来开发该系统。观察者模式的类图如图 5-1 所示。

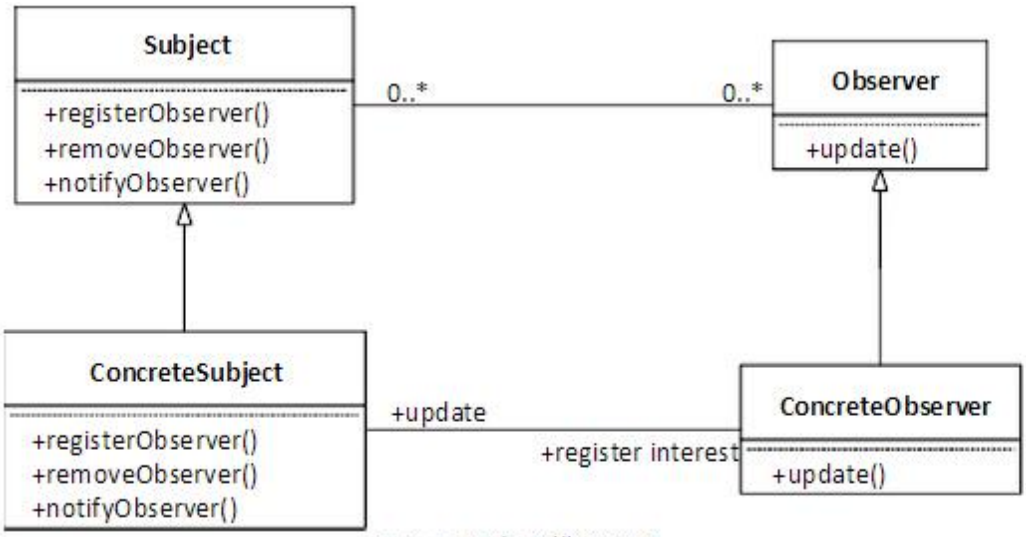


图 5-1 观察者模式类图

【C++代码】

```
#include <iostream>
#include <vector>
using namespace std;
class Observer {
public:
  virtual void update(float temp, float humidity, float cleanness)=0;
};
class Subject {
public:
  virtual void registerObserver(Observer *o)=0;  //注册对主题感兴趣的观察者
  virtual void removeObserver(Observer *o)=0;  //删除观察者
  virtual void notifyObservers()=0; //当主题发生变化时通知观察者
};
class EnvironmentData:public   (1)   {
private:
   vector<Observer *> observers;
    float temperature,humidity,cleanness;
 public:
    void registerObserver(Observer* o) {      observers.push_back(o);   }
    void removeObserver(Observer* o) {   /*代码省略 */  }
    void notifyObservers(){
    for(vector<Observer*>::const_iterator it = observers.begin();it!=observers.e
    { ____(2)____ }
}
```

```
void measurementsChanged(){     (3)     }
void setMeasurements(float temperature,float humidity,float cleanness) {
     this->temperature = temperature;
     this->humidity = humidity;
     this->cleanness = cleanness;
        (4)

    }
};
class CurrentConditionsDisplay:public   (5)   {
private:
      float temperature,humidity,cleanness;
      Subject* envData;
public:
     CurrentConditionsDisplay(Subject* envData) {
         this->envData = envData;
          (6)
     }
     void update(float temperature,float humidity,float cleanness) {
     this->temperature = temperature;
     this->humidity = humidity;
     this->cleanness = cleanness;
     display();
}
void display() {   /*代码省略 */ }
};
int main() {
     EnvironmentData* envData = new EnvironmentData();
     CurrentConditionsDisplay* currentDisplay = new CurrentConditionsDisplay(envData);
     envData->setMeasurements(80,65,30.4f);
     return 0;
}
```

【问题 1】

阅读说明和 Java 代码，将应填入（n）处的字句写在答题纸的对应栏内。

【参考答案】

1）Subject

2）(*it)->update（remperature，humidity，cleanness）

3）notify()bservesers

4）mesasurementsChanged()

5）Observer

6）envData->regisiterObserver（this）

【试题分析】

本题考察观察者（Observer)模式的概念及应用。

观察者模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

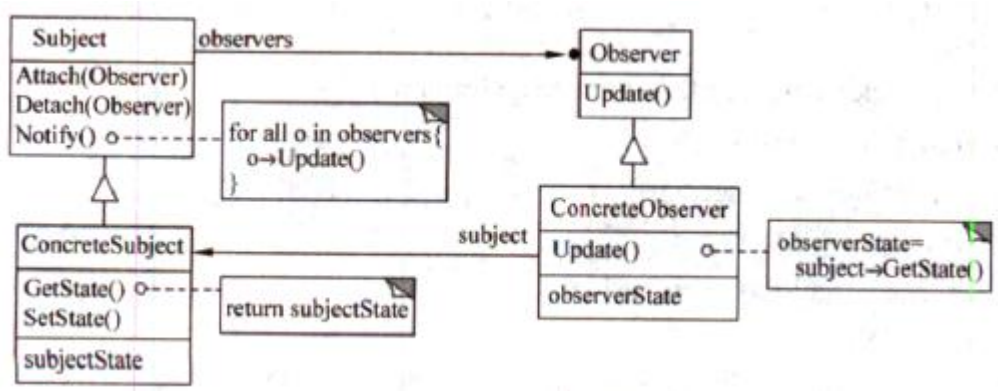
Observer 模式适用于：

①当一个抽象模型有两个方面，其中一个方面依赖于另一个方面。将这两者封装在独立地对象中以使它们可以各自独立地改变和复用。

②当对一个对象的改变需要同时改变其他对象，而不知道具体有多少对象有待改变时。

③当一个对象必须通知其他对象，而它又不能假定其他对象是谁，即：不希望这些对象是紧耦合的。

观察者模式的结构如下图所示，其中：



Subject（主题）知道它的观察者，可以有任意多个观察者观察同一个目标；提供注册和删除观察者对象的接口。

Observer（观察者）为那些在目标发生改变时需获得通知的对象定义一个更新接口。

ConcreteSubject（具体主题）将有关状态存入一个 ConcreteObserver 对象；当它的状态发生改变时，向它的各个观察者发出通知。

ConcreteObserver（具体观察者）维护一个指向 ConcreteSubject 对象的引用；存储有关状态，这些状态应与主题的状态保持一致；实现 Observer 的更新接口以使自身状态与主题的状态保持一致。

在本题的说明中，给出了观察者模式的结构图，答题时需要首先确定程序中的类与观察者模式结构的对应关系，也就是要找到哪些是 ConcreteSubject，哪些是 ConcreteObserver。由程序上下文可以判断出，类 EnvironmentData 对应的是 ConcreteSubject，类 CurrentConditionsDisplay 对应的是 ConcreteObserver。根据类图，它们分别为 Subject 和 Observer 的子类。由此可以，空（1)和空（5)应分别填写 Subject 和 Observer。

空（2)要求给出方法 notifyObservers 的实现，其功能是在主题发生变化时通知观察者。通

知的实现是通过向对该主题感兴趣的所有观察者发送 update 消息实现的。对主题 EnvironmentData 感兴趣的观察者由向量 observers 表示,所以在 notifyObservers 方法中,就是对向量 observers 中的每个成员发送 update 消息,因此空(2)应填写(*it)->update(temperature, humidity, cleanness)。

方法 measurementsChanged 表示主题发生了变化,这时应该通知对应的观察者,所以空(3)处应填写 notifyObservers()。

方法 setMeasurements 用于设置发生变化后的主题内容,所以空(4)处应填写 measurementsChanged()。

CurrentConditionsDisplay 是对主题 EnvironmentData 感兴趣的一个观察者,要能够获得主题的变化,需要首先将自己注册为该主题的观察者,这个注册行为在其构造函数中完成。因此空(6)处应填写 envData->registerObserver(this)。

试题六（共 15 分）

【说明】

某实验室欲建立一个实验室环境监测系统，能够显示实验室的温度、湿度以及洁净度等环境数据。当获取到战新的环境测量数据时，显示的环境数据能够更新。

现在采用观察者（Observer)模式来开发该系统。观察者模式的类图如图 6-1 所示。

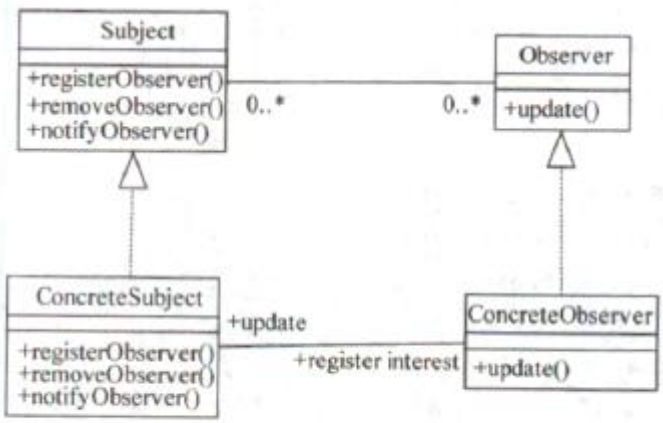


图 6-1 观察者模式类图

【Java 代码】

```
import java.util.*;
interface Observer {
  public void update(float temp, float humidity, float cleanness);
}
interface Subject {
  public void registerObserver(Observer o); //注册对主题感兴趣的观察者
  public void removeObserver(Observer o);   //删除观察者
  public void notifyObservers();            //当主题发生变化时通知观察者
}
class EnvironmentData implements   (1)   {
  private ArrayList observers;
  private float temperature, humidity, cleanness;
  public EnvironmentData() {   observers = new ArrayList();  }
  public void registerObserver(Observer o) {  observers.add(o);  }
  public void removeObserver(Observer o) {  /* 代码省略 */  }
  public void notifyObservers() {
    for (int i = 0; i < observers.size(); i++) {
      Observer observer = (Observer)observers.get(i);
      (2) ;
    }
  }
```

```
  public void measurementsChanged() { ___(3)___; }
public void setMeasurements(float temperature, float humidity, float
cleanness) {
    this.temperature = temperature;
    this.humidity = humidity;
    this.cleanness = cleanness;
    ___(4)___;
  }
}
class CurrentConditionsDisplay implements ___(5)___{
    private float temperature;
    private float humidity;
    private float cleanness;
    private Subject envData;
    public CurrentConditionsDisplay(Subject envData) {
      this.envData = envData;
      ___(6)___;
}
    public void update(float temperature, float humidity, float cleanness) {
      this.temperature = temperature;
      this.humidity = humidity;
      this.cleanness = cleanness;
      display();
}
    public void display() { /* 代码省略 */ }
}
class EnvironmentMonitor{
  public static void main(String[] args) {
    EnvironmentData envData = new EnvironmentData();
    CurrentConditionsDisplay currentDisplay = new CurrentConditions
    Display(envData);
    envData.setMeasurements(80, 65, 30.4f);
  }
}
```

【问题 1】

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

【参考答案】

(1) Subject

(2) observer.update(temperature, humidity, cleanness)

(3) notifyObservers()

(4) measurementsChanged()

(5) Observer

(6) envData.registerObserver(this)

【试题分析】

本题考查观察者（Observer)模式的概念及应用。

观察者模式定义对象间的一种一对多的依赖关系，当一个对象的状态发生改变时，所有依赖于它的对象都得到通知并被自动更新。

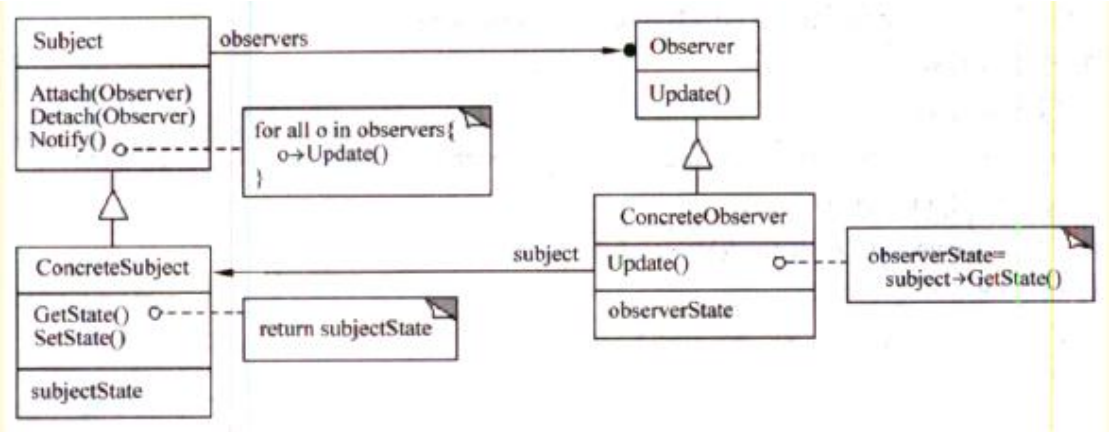
Observer 模式适用于：

①当一个抽象模型有两个方面，其中一个方面依赖于另一个方面。将这两者封装在独立的对象中以使它们可以各自独立地改变和复用。

②当对一个对象的改变需要同时改变其他对象，而不知道具体有多少对象有待改变时。

③当一个对象必须通知其他对象，而它又不能假定其他对象是谁，即：不希望这些对象是紧耦合的。

观察者模式的结构如下图所示，其中：



Subject（主题）知道它的观察者，可以有任意多个观察者观察同一个目标；提供注册和删除观察者对象的接口。

Observer（观察者）为那些在目标发生改变时需获得通知的对象定义一个更新接口。

ConcreteSubject（具体主题）将有关状态存入一个 ConcreteObserver 对象；.当它的状态发生改变时，向它的各个观察者发出通知。

ConcreteObserver（具体观察者）维护一个指向 ConcreteSubject 对象的引用；存储有关状

态，这些状态应与主题的状态保持一致；实现 Observer 的更新接口以使自身状态与主题的状态保持一致。

在本题的说明中，给出了观察者模式的结构图，答题时需要首先确定程序中的类与观察者模式结构的对应关系，也就是要找到哪些是 ConcreteSubject,哪些是 ConcreteObserver。由程序上下文可以判断出，类 EnvironmentData 对应的是 ConcreteSubject，类 CurrentConditionsDisplay 对应的是 ConcreteObserver。根据类图，它们分别为 Subject 和 Observer 的子类。由此可以，空（1)和空（5)应分别填写 Subject 和 Observer。

空（2)要求给出方法 notifyObservers 的实现，其功能是在主题发生变化时通知观察者。通知的实现是通过向对该主题感兴趣的所有观察者发送 update 消息实现的。对主题 EnvironmentData 感兴趣的观察者由向量 observers 表示,所以在 notifyObservers 方法中，就是对向量 observers 中的每个成员发送 update 消息，因此空（2)应填写 observer.update(temperature，humidity, cleanness)。

方法 measurementsChanged 表示主题发生了变化，这时应该通知对应的观察者，所以空（3)处应填写 notifyObservers()。

方法 setMeasurements 用于设置发生变化后的主题内容，所以空（4)处应填写 measurementsChanged()。

CurrentConditionsDisplay 是对主题 EnvironmentData 感兴趣的一个观察者，要能够获得主题的变化，需要首先将自己注册为该主题的观察者，这个注册行为在其构造函数中完成。因此空（6)处应填写 envData.registerObserver(this)。