

在汇编指令中，操作数在某寄存器中的寻址方式称为(1)寻址。

- (1) A. 直接                      B. 变址                      C. 寄存器                      D. 寄存器间接

**【答案】C**

**【解析】**

立即寻址：是一种特殊的寻址方式，指令中在操作码字段后面的部分不是通常意义上的操作数地址，而是操作数本身，也就是说数据就包含在指令中，只要取出指令，也就取出了可以立即使用的操作数。

直接寻址：在直接寻址中，指令中地址码字段给出的地址 A 就是操作数的有效地址，即形式地址等于有效地址。

间接寻址：间接寻址意味着指令中给出的地址 A 不是操作数的地址，而是存放操作数地址的主存单元的地址，简称操作数地址的地址。

寄存器寻址：寄存器寻址指令的地址码部分给出了某一个通用寄存器的编号 Ri，这个指定的寄存器中存放着操作数。

寄存器间接寻址：在寄存器间接寻址方式中，寄存器内存放的是操作数的地址，而不是操作数本身，即操作数是通过寄存器间接得到的。

变址寻址：变址寻址就是把变址寄存器 Rx 的内容与指令中给出的形式地址 A 相加，形成操作数有效地址，即  $EA = (R_x) + A$ 。

基址寻址：基址寻址是将基址寄存器 Rb 的内容与指令中给出的位移量 D 相加，形成操作数有效地址，即  $EA = (R_b) + D$ 。

相对寻址：相对寻址是基址寻址的一种变通，由程序计数器提供基准地址，指令中的地址码字段作为位移量 D，两者相加后得到操作数的有效地址，即  $EA = (PC) + D$ 。

计算机系统中，虚拟存储体系由(2)两级存储器构成。

- (2) A. 主存—辅存              B. 寄存器—Cache              C. 寄存器—主存              D. Cache—主存

**【答案】A**

**【解析】**

虚拟存储器：具有部分装入对换功能，能从逻辑上对内存容量进行大幅度扩充，使用方便的一种存储器系统。虚拟存储器的容量与主存大小无关。虚拟存储器的基本思路是：作业提交时，先全部部进入辅助存储器，作业投入运行时，不把作业的全部信息同时装入主存储器，而是将其中当前使用部分先装入主存储器，其余暂时不用的部分先存放在作为主存扩充

的辅助存储器中，待用到这些信息时，再由系统自动把它们装入到主存储器中。

程序计数器(PC)是(3)中的寄存器。

- (3) A. 运算器                      B. 控制器                      C. Cache                      D. I/O 设备

**【答案】B**

**【解析】**

运算器由算术逻辑单元 (ALU)、累加寄存器、数据缓冲寄存器和状态条件寄存器组成，是数据加工的处理部件，完成计算机的各种算术和逻辑运算。

控制器用于控制整个 CPU 的工作，决定了计算机运行过程的自动化，不仅要保证程序的正确执行，而且要能够处理异常的事件。控制器包含：程序计数器 (PC)、指令寄存器 (IR)、地址寄存器 (AR)、指令译码器 (ID)、时序部件等。

中断向量提供(4)。

- (4) A. 外设的接口地址                      B. 待传送数据的起始和终止地址  
C. 主程序的断点地址                      D. 中断服务程序入口地址

**【答案】D**

**【解析】**

中断是指在计算机执行程序的过程中，当出现异常情况或者特殊请求时，计算机停止现行的程序的运行，转而对这些异常处理或者特殊请求的处理，处理结束后再返回到现行程序的中断处，继续执行原程序。

中断向量：中断服务程序的入口地址。

中断向量表：把系统中所有的中断类型码及对应的中断向量按一定的规律存放在一个区域内，这个存储区域就称为中断向量表。

CPU 是根据中断号获取中断向量值，即对应中断服务程序的入口地址值。

在计算机系统中总线宽度分为地址总线宽度和数据总线宽度。若计算机中地址总线的宽度为 32 位，则最多允许直接访问主存储器(5)的物理空间。

- (5) A. 40MB                      B. 4GB                      C. 40GB                      D. 400GB

**【答案】B**

**【解析】**

本题考查2进制计算的基础知识。

其实题目里的总线宽度是背景信息，与整个考察的计算关系不大，地址总线的位数决定了CPU可直接寻址的内存空间大小，数据总线宽度就是你一次传输的数据是多少位的。2进制的32位换算成物理空间的计算为： $2^{32}B = 2^{22}KB = 2^{12}MB = 2^2GB = 4GB$

为了提高计算机磁盘存取效率，通常可以(6)。

- (6) A. 利用磁盘格式化程序，定期对ROM进行碎片整理
- B. 利用磁盘碎片整理程序，定期对内存进行碎片整理
- C. 利用磁盘碎片整理程序，定期对磁盘进行碎片整理
- D. 利用磁盘格式化程序，定期对磁盘进行碎片整理

【答案】C

【解析】

格式化程序使用于磁盘格式化；格式化是指对磁盘或磁盘中的分区（partition）进行初始化的一种操作，这种操作通常会导致现有的磁盘或分区中所有的文件被清除。

碎片整理程序用于磁盘碎片整理；磁盘碎片整理，就是通过系统软件或者专业的磁盘碎片整理软件对电脑磁盘在长期使用过程中产生的碎片和凌乱文件重新整理，可提高电脑的整体性能和运行速度。

磁盘碎片应该称为文件碎片，是因为文件被分散保存到整个磁盘的不同地方，而不是连续地保存在磁盘连续的簇中形成的。硬盘在使用一段时间后，由于反复写入和删除文件，磁盘中的空闲扇区会分散到整个磁盘中不连续的物理位置上，从而使文件不能存在连续的扇区里。这样，再读写文件时就需要到不同的地方去读取，增加了磁头的来回移动，降低了磁盘的访问速度。

内存是随机访问存取，文件在任何位置读取的时间是一样的。

安全的电子邮件协议为(7)。

- (7) A. MIME
- B. PGP
- C. POP3
- D. SMTP

【答案】B

【解析】

SMTP（Simple Mail Transfer Protocol）即简单邮件传输协议：用于电子邮件的传递和投递。

POP3 (Post Office Protocol - Version 3) 即邮局协议版本 3: 用于支持使用客户端远程管理在服务器上的电子邮件, 是一种离线的收邮件的协议。

MIME (Multipurpose Internet Mail Extensions) 即多用途互联网邮件扩展类型: 它设计的最初目的是为了在发送电子邮件时附加多媒体数据, 让邮件客户程序能根据其类型进行处理。当被 HTTP 协议支持之后, 它的意义就更为显著了。它使得 HTTP 传输的不仅是普通的文本, 而变得丰富多彩。

PGP (Pretty Good Privacy) 即更好地保护隐私; 是一个基于 RSA 公钥加密体系的邮件加密软件。可以用它对邮件保密以防止非授权者阅读, 它还能对邮件加上数字签名从而使收信人可以确认邮件的发送者, 并能确信邮件没有被篡改。它可以提供一种安全的通讯方式, 而事先并不需要任何保密的渠道用来传递密钥。它采用了一种 RSA 和传统加密的杂合算法, 用于数字签名的邮件文摘算法, 加密前压缩等, 还有一个良好的人机工程设计。它的功能强大, 有很快的速度。

操作系统通过 (8) 来组织和管理外存中的信息。

- (8) A. 字处理程序      B. 设备驱动程序      C. 文件目录和目录项      D. 语言翻译程序

**【答案】C**

**【解析】**

字处理程序: 用于文字的格式化和排版, 文字处理软件的发展和文字处理的电子化是信息社会发展的标志之一。

设备驱动程序: 是一种可以使计算机和设备通信的特殊程序。相当于硬件的接口, 操作系统只有通过这个接口, 才能控制硬件设备的工作。

语言翻译程序: 是一种系统程序, 它将计算机编程语言编写的程序翻译成另外一种计算机语言等价的程序, 主要包括编译程序和解释程序, 汇编程序也被认为是翻译程序。

下列操作系统中, (9) 保持网络系统的全部功能, 并具有透明性、可靠性和高性能等特性。

- (9) A. 批处理操作系统      B. 分时操作系统      C. 分布式操作系统      D. 实时操作系统

**【答案】C**

**【解析】**

分布式操作系统是网络操作系统的更高级形式, 保持网络系统所拥有的全部功能, 同时又

有透明性、可靠性和高性能等。

在进程状态转换过程中，可能会引起进程阻塞的原因是 (10)。

- (10) A. 时间片到      B. 执行 V 操作      C. I/O 完成      D. 执行 P 操作

**【答案】D**

**【解析】**

运行态：占有处理器正在运行

就绪态：指具备运行条件，等待系统分配处理器以便运行

等待态：又称为阻塞态或睡眠态，指不具备运行条件，正在等待某个

运行态——等待态：等待使用资源，如等待外设传输，等待人工干预

等待态——就绪态：资源得到满足，如外设传输结束，人工干预完成

运行态——就绪态：运行时间片到，出现有更高优先权进程

就绪态——运行态：CPU 空闲时选择一个就绪进程

本题中，时间片到，进程应该进入就绪态；I/O 完成进程应该是阻塞态到就绪态；V 操作是释放资源，到一个进程释放资源，应该会唤醒另一个进程运行。所以最适合的选项应为 D

假设系统有  $n$  个进程共享资源  $R$ ，且资源  $R$  的可用数为 3，其中  $n \geq 3$ 。若采用 PV 操作，则信号量  $S$  的取值范围应为 (11)。

- (11) A.  $-1 \sim n-1$       B.  $-3 \sim 3$       C.  $-(n-3) \sim 3$       D.  $-(n-1) \sim 1$

**【答案】C**

**【解析】**

PV 操作：是实现进程同步和互斥的常用方法，P 操作和 V 操作是低级通信原语，在执行期间不可分割；其中 P 操作表示申请一个资源，V 操作表示释放一个资源。

P 操作的定义： $S := S - 1$ ，若  $S \geq 0$ ，则执行 P 操作的进程继续执行；若  $S < 0$ ，则将该进程设为阻塞状态（因为无可用资源），并将其插入阻塞队列。

V 操作的定义： $S := S + 1$ ，若  $S > 0$ ，则执行 V 操作的进程继续执行；若  $S \leq 0$ ，则从阻塞状态唤醒一个进程，并将其插入就绪队列，然后执行 V 操作的进程继续。

本题中  $S$  初始值为 3，当  $n$  个进程同时执行时，需要执行  $n$  次 P 操作，这时信号量的值应为  $3-n$ ，所以信号量的变化范围为： $-(n-3) \sim 3$

已知函数  $f()$ 、 $g()$  的定义如下所示，调用函数  $f$  时传递给形参  $x$  的值是 5。若  $g(a)$  采用引用调用 (callbyreference) 方式传递参数，则函数  $f$  的返回值为 (12)；若  $g(a)$  采用值调用 (callbyvalue) 的方式传递参数，则函数  $f$  的返回值为 (13)。其中，表达式 “ $x \gg 1$ ” 的含义是将  $x$  的值右移 1 位，相当于  $x$  除以 2。

$f(int\ x)$	$g(int\ x)$
<pre>int a = x &gt;&gt; 1; g(a); return a + x;</pre>	<pre>x = x * (x + 1); return;</pre>

- (12) A. 35                      B. 32                      C. 11                      D. 7
- (13) A. 35                      B. 32                      C. 11                      D. 7

**【答案】C D**

**【解析】**

传值调用：形参取的是实参的值，形参的改变不会导致调用点所传的实参的值发生改变。

引用（传址）调用：形参取的是实参的地址，即相当于实参存储单元的地址引用，因此其值的改变同时也就改变了实参的值。

本题中， $a = x \gg 1$ ， $x$  值为 5，则  $a = 2$ ；采用传值调用时，由于  $g(a)$  对数据  $x$  的运算不会影响原来  $x$  的值，所以  $a + x = 5 + 2 = 7$ ；采用传址调用时，由于  $g(a)$  对数据  $x$  的运算会影响原来  $x$  的值，当执行  $g(a)$  后  $x$  的值为 6，所以  $a + x = 5 + 6 = 11$ 。

设数组  $a[0..n-1, 0..m-1]$  ( $n > 1, m > 1$ ) 中的元素以行为主序存放，每个元素占用 4 个存储单元，则数组元素  $a[i, j]$  ( $0 \leq i < n, 0 \leq j < m$ ) 的存储位置相对于数组空间首地址的偏移量为 (14)。

- (14) A.  $(j * m + i) * 4$       B.  $(i * m + j) * 4$       C.  $(j * n + i) * 4$       D.  $(i * n + j) * 4$

**【答案】B**

**【解析】**

由于  $a[i, j]$  ( $0 \leq i < n, 0 \leq j < m$ )  $i$  和  $j$  是从 0 开始的，以行为主序，则  $a[i, j]$  前面已经有  $i$  行已经排列满，一共有  $i * m$  个元素，在  $a[i, j]$  元素所在的行，前有  $j$  个元素，所以  $a[i, j]$  前一共有  $(i * m + j)$  个元素；每个元素占用 4 个存储单元，所以  $a[i, j]$  相对数组空间首地址的偏移量为  $(i * m + j) * 4$ 。

线性表采用单循环链表存储的主要优点是(15)。

- (15)A. 从表中任一结点出发都能遍历整个链表  
B. 可直接获取指定结点的直接前驱和直接后继结点  
C. 在进行删除操作后，能保证链表不断开  
D. 与单链表相比，更节省存储空间

**【答案】A**

**【解析】**

循环链表：在单向链表（或双向链表的基础上），令表尾节点的指针指向表中的第一个节点，构成循环链表。其特点是可以从表中任意节点开始遍历整个链表。

若某线性表长度为  $n$  且采用顺序存储方式，则运算速度最快的操作是(16)。

- (16)A. 查找与给定值相匹配的元素的位置  
B. 查找并返回第  $i$  个元素的值 ( $1 \leq i \leq n$ )  
C. 删除第  $i$  个元素 ( $1 \leq i < n$ )  
D. 在第  $i$  个元素 ( $1 \leq i \leq n$ ) 之前插入一个新元素

**【答案】B**

**【解析】**

在线性表中插入和删除元素都需要修改前驱和后继的指针。

查找并返回第  $i$  个元素的值，这个只要找到该位置读取即可。

查找与给定值相匹配的元素的位置，先读取第一个元素再比较，依次类推直到找到该元素。

设元素  $a$ 、 $b$ 、 $c$ 、 $d$  依次进入一个初始为空的栈，则不可能通过合法的栈操作序列得到(17)。

- (17)A.  $a\ b\ c\ d$                       B.  $b\ a\ c\ d$                       C.  $c\ a\ b\ d$                       D.  $d\ c\ b\ a$

**【答案】C**

**【解析】**

选项 A:  $a$  进栈、 $a$  出栈、 $b$  进栈、 $b$  出栈、 $c$  进栈、 $c$  出栈、 $d$  进栈、 $d$  出栈

选项 B:  $a$  进栈、 $b$  进栈、 $b$  出栈、 $a$  出栈、 $c$  进栈、 $c$  出栈、 $d$  进栈、 $d$  出栈

选项 C: 无法实现

选项 D: a 进栈、b 进栈、c 进栈、d 进栈、d 出栈、c 出栈、b 出栈、a 出栈

若要求对大小为  $n$  的数组进行排序的时间复杂度为  $O(n\log_2 n)$ ，且是稳定的（即如果待排序的序列中两个数据元素具有相同的值，在排序前后它们的相对位置不变），则可选的排序方法是 (18)。

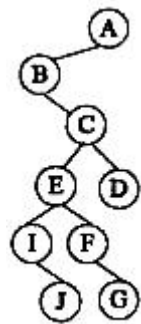
- (18) A. 快速排序                      B. 归并排序                      C. 堆排序                      D. 冒泡排序

【答案】B

【解析】

类 别	排序方法	时间复杂度		空间复杂度	稳 定 性
		平均情况	最坏情况	辅助存储	
插入排序	直接插入	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
	Shell排序	$O(n^{1.3})$	$O(n^2)$	$O(1)$	不稳定
选择排序	直接选择	$O(n^2)$	$O(n^2)$	$O(1)$	不稳定
	堆排序	$O(n\log_2 n)$	$O(n\log_2 n)$	$O(1)$	不稳定
交换排序	冒泡排序	$O(n^2)$	$O(n^2)$	$O(1)$	稳定
	快速排序	$O(n\log_2 n)$	$O(n^2)$	$O(\log_2 n)$	不稳定
归并排序		$O(n\log_2 n)$	$O(n\log_2 n)$	$O(n)$	稳定
基数排序		$O(d(r+n))$	$O(d(r+n))$	$O(r+n)$	稳定

对于一般的树结构，可以采用孩子一兄弟表示法，即每个结点设置两个指针域，一个指针（左指针）指示当前结点的第一个孩子结点，另一个指针（右指针）指示当前结点的下一个兄弟结点。某树的孩子一兄弟表示如下图所示。以下关于结点 D 与 E 的关系的叙述中，正确的是 (19)。



- (19) A. 结点 D 与结点 E 是兄弟                      B. 结点 D 是结点 E 的祖父结点  
C. 结点 E 的父结点与结点 D 的父结点是兄弟    D. 结点 E 的父结点与结点 D 是兄弟

【答案】D

【解析】

孩子兄弟表示法的每个节点有两个指针域，一个指向其长子，另一个指向其兄弟。



某企业研发信息系统的过程中 (20) 不属于数据库管理员 (DBA) 的职责。

- (20) A. 决定数据库中的信息内容和结构
- B. 决定数据库的存储结构和存取策略
- C. 进行信息系统程序的设计和编写
- D. 定义数据的安全性要求和完整性约束条件

【答案】C

【解析】

数据库管理员 (DBA): 负责数据库的总体信息控制。具体职责包括: 决定数据库中信息内容和结构; 决定数据库的存储结构和存取策略; 定义数据库的安全性要求和完整性约束条件; 监控数据库的使用和运行; 数据库的性能改进、数据库的重组和重构, 以提高系统的性能。

某高校人事管理系统中, 规定讲师每课时的教学酬金不能超过 100 元, 副教授课时的 学酬金不能超过 130 元, 教授每课时的教学酬金不能超过 160 元。这种情况下所设置的数据完整性约束条件称之为 (21)。

- (21) A. 用户定义完整性      B. 实体完整性      C. 主键约束完整性      D. 参照完整性

【答案】A

【解析】

完整性约束: 防止的是对数据的意外破坏。

实体完整性: 规定基本关系 R 的主属性 A 不能取空。

用户自定义完整性: 就是针对某一具体关系数据库的约束条件, 反映某一具体应用所涉及的数据必须满足的语义要求, 由应用的环境决定。如: 年龄必须为大于 0 小于 150 的整数。

参照完整性/引用完整性: 规定, 若 F 是基本关系 R 的外码, 它与基本关系 S 的主码 K, 相对应 (基本关系 R 和 S 不一定是不同的关系), 则 R 中每个元组在 F 上的值必须为: 或者取空值; 或者等于 S 中某个元组的主码值。

本题中是按照业务系统自身的要求来定义数据的约束, 属于用户自定义完整性。

某教学管理数据库中, 学生、课程关系模式和主键分别为: S (学号, 姓名, 性别, 家庭住址, 电话), 关系 S 的主键为学号; C (课程号, 课程名, 学分), 关系 C 的主键为课程

号。假设一个学生可以选择多门课程，一门课程可以由多个学生选择。一旦学生选择某门课程必定有该课程的成绩。由于学生与课程之间的“选课”联系类型为(22)所以对该联系(23)。

(22)A. n:m                      B. 1:n                      C. n:1                      D. 1:1

(23)A. 不需要构建一个独立的关系模式

B. 需要构建一个独立的关系模式，且关系模式为：SC（课程号，成绩）

C. 需要构建一个独立的关系模式，且关系模式为：SC（学生号，成绩）

D. 需要构建一个独立的关系模式，且关系模式为：SC（学生号，课程号，成绩）

**【答案】A D**

**【解析】**

一个学生可以选择多门课程，一门课程可以由多个学生选择，说明学生与选课之间的联系类型为多对多。

对于多对多的联系转换成关系时，应转为一个独立的关系，联系的属性由两端实体的码和联系的属性组成。该关系码为两端实体集码共同组成。对于本题来说 联系本身需要记录成绩，所以 SC 关系应该由学生号、课程号、成绩三个属性组成。

查询“软件工程”课程的平均成绩、最高成绩与最低成绩之间差值的 SQL 语句如下：

SELECT AVG(成绩)AS 平均成绩, (24)

FROM C, SC

WHERE C. 课程名=' 软件工程' AND C. 课程号=SC. 课程号;

(24)A. 差值 AS MAX（成绩）— MIN（成绩）

B. MAX（成绩）— MIN（成绩）AS 差值

C. 差值 IN MAX（成绩）— MIN（成绩）

D. MAX（成绩）— MIN（成绩）IN 差值

**【答案】B**

**【解析】**

给列取别名的语法为： 列名 AS 新列名；

最大值聚集函数为：MAX；最小值聚集函数为：MIN

能隔离局域网中广播风暴、提高带宽利用率的设备是(25)。

(25)A. 网桥                      B. 集线器                      C. 路由器                      D. 交换机

【答案】C

【解析】

路由器可以分割广播风暴；交换机可以分割冲突域；

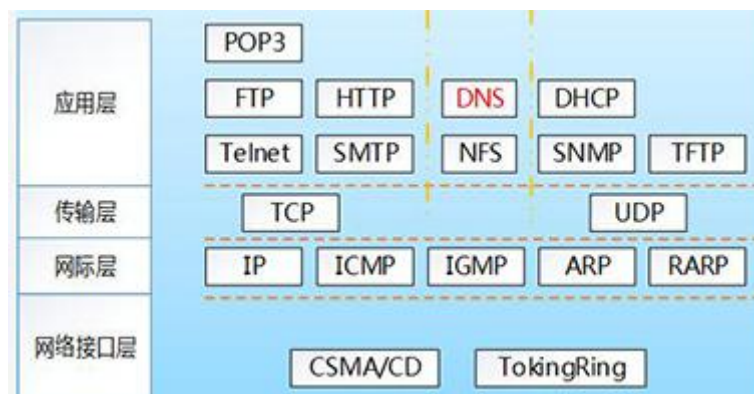
下面的协议中属于应用层协议的是 (26)，该协议的报文封装在 (27) 中传送。

(26) A. SNMP                      B. ARP                      C. ICMP                      D. X. 25

(27) A. TCP                      B. IP                      C. UDP                      D. ICMP

【答案】A    C

【解析】



某公司内部使用 wb. xyz. com. cn 作为访问某服务器的地址，其中 wb 是 (28)。

(28) A. 主机名                      B. 协议名                      C. 目录名                      D. 文件名

【答案】A

【解析】

wb. xyz. com. cn 中：wb 为主机名；xyz. com. cn 为域名。

如果路由器收到了多个路由协议转发的关于某个目标的多条路由，那么决定采用哪条路由的策略是 (29)。

(29) A. 选择与自己路由协议相同的                      B. 选择路由费用最小的  
C. 比较各个路由的管理距离                      D. 比较各个路由协议的版本

【答案】C

【解析】

管理距离决定了路由的优先，管理距离越小说明路由优先级更高。

(30) A. 220. 112. 145. 32/22                      B. 220. 112. 145. 64/22  
C. 220. 112. 147. 64/22                      D. 220. 112. 177. 64/22

**【解析】**

11111111.11111111.11111111	00.00000000	/22
11011100.011100000.101100	11.01011100	220.112.179.92
11011100.011100000.101100	01.01000000	220.112.177.64
11011100.011100000.100100	11.01000000	220.112.147.64
11011100.011100000.100100	01.01000000	220.112.145.64
11011100.011100000.100100	01.00100000	220.112.145.32

某开发小组欲开发一个软件系统,实现城市中不同图书馆的资源共享,包括实体资源和电子资源,共享规则可能在开发过程中有变化。客户希望开发小组能尽快提交可运行的软件,且可以接受多次交付。这种情况下最适宜采用(31)开发过程模型。主要是因为这种模型(32)。

(31) A. 瀑布                      B. 原型                      C. 增量                      D. 螺旋

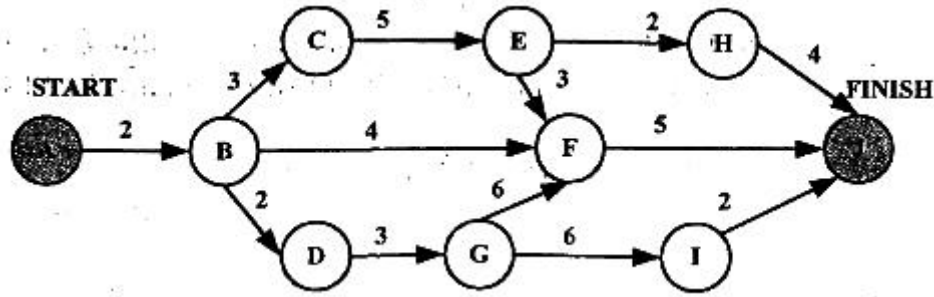
(32) A. 可以快速提交阶段性的软件产品                      B. 需求变化对开发没有影响

         C. 减少用户适应和习惯系统的时间和精力                      D. 能够很好解决风险问题

**【解析】**

题干中明确说明希望快速开发，同时可以接受多次交互。这种情况下适合增量模型。这样可以快速开发第一交互产品、交互，然后再开发、再交互。

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时间（天），则完成该项目的最少时间为33天。活动BC和BF分别最多可以晚开始（34）天而不会影响整个项目的进度。



- (33) A. 11                      B. 15                      C. 16                      D. 18
- (34) A. 0 和 7                      B. 0 和 11                      C. 2 和 7                      D. 2 和 11

【答案】D A

【解析】

本题中关键路径为：A→B→D→G→F→J 和 A→B→C→E→F→J 总时长为：18

所以完成该项目的最小时间为 18 天

BC：由于 BC 在关键路径上，所以 BC 的松弛时间为 0

BF：由于关键路径时长为 18，经过 BF 的最长路路径 11，所以 BF 松弛时间为：18-11=7

结构化开发方法中，(35) 主要包含对数据结构和算法的设计。对算法设计时，其主要依据来自 (36) 描述算法时，(37) 不是理想的表达方式。

- (35) A. 体系结构设计                      B. 数据设计                      C. 接口设计                      D. 过程设计
- (36) A. 数据流图                      B. E-R 图                      C. 状态-迁移图                      D. 加工规格说明
- (37) A. 流程图                      B. 决策图                      C. 程序设计语言代码                      D. 伪代码

【答案】D D A

【解析】

过程设计主要包含对数据结构和算法的设计。

状态—迁移图(STD)或状态—迁移表被用来描述系统或对象的状态，以及导致系统或对象的状态改变的事件，从而描述系统的行为。

数据流图是结构化分析方法中使用的工具,它以图形的方式描绘数据在系统中流动和处理的过程,由于它只反映系统必须完成的逻辑功能,所以它是一种功能模型。在结构化开发方法中,数据流图是需求分析阶段产生的结果。

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。

算法可以借助各种工具描述出来，一个算法可以用自然语言、数字语言或约定的符合来描述，也可以用计算机高级程序语言来描述，如流程图、Pascal 语言、C 语言、伪代码或决策表等。

模块 A 的功能为：从数据库中读出产品信息，修改后存回数据库，然后将修改记录写到维护文件中。该模块内聚类型为 (38) 内聚。以下关于该类内聚的叙述中，正确的是 (39)。

- (38) A. 逻辑                      B. 时间                      C. 过程                      D. 功能
- (39) A. 是最低的内聚类型                      B. 是最高内聚类型
- C. 不易于重用                      D. 模块独立性好

【答案】C    C

【解析】

偶然聚合：模块完成的动作之间没有任何关系，或者仅仅是一种非常松散的关系。

逻辑聚合：模块内部的各个组成在逻辑上具有相似的处理动作，但功能用途上彼此无关。

时间聚合：模块内部的各个组成部分所包含的处理动作必须在同一时间内执行。

过程聚合：模块内部各个组成部分所要完成的动作虽然没有关系，但必须按特定的次序执行。

通信聚合：模块的各个组成部分所完成的动作都使用了同一个数据或产生同一输出数据。

顺序聚合：模块内部的各个部分，前一部分处理动作的最后输出是后一部分处理动作的输入。

功能聚合：模块内部各个部分全部属于一个整体，并执行同一功能，且各部分对实现该功能都必不可少

本题中模块 A 内部的各个部分处理成分是需要按照特定的次序来执行的，结合题干的选择项来看，选择过程聚合比较符合题意，这种聚合不利于模块的重用

某个应用中，需要对输入数据进行排序，输入数据序列基本有序（如输入为 1, 2, 5, 3, 4, 6, 8, 7）。在这种情况下，采用 (40) 排序算法最好，时间复杂度为 (41)。

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

【答案】A    A

【解析】

当序列基本有序时。使用插入排序效率是最高的，能达到这种算法的最优效果， $O(n)$

在结构化分析中，用数据流图描述(42)。当采用数据流图对银行客户关系管理进行分析时，(43)是一个加工。

- (42)A. 数据对象之间的关系，用于对数据建模  
B. 数据在系统中如何被传送或变换，以及如何对数据流进行变换  
C. 系统对外部事件如何响应，如何动作，用于对行为建模  
D. 系统中的数据对象和控制信息的特性

- (43)A. 工作人员                      B. 账户                      C. 余额                      D. 存款

【答案】B    C

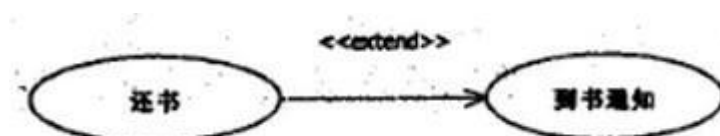
【解析】

数据流图 (Data Flow Diagram, DFD): 是一种最常用的结构化分析工具，从数据传递和加工的角度，以图形的方式刻画系统内数据的运动情况。DFD 摆脱了系统的物理内容，精确地在逻辑上描述系统的功能、输入、输出和数据存储等，是系统逻辑模型的重要组成部分。加工: 描述了输入数据流到输出数据流之间的变换，也就是输入数据流经过什么处理后变成了输出数据流。

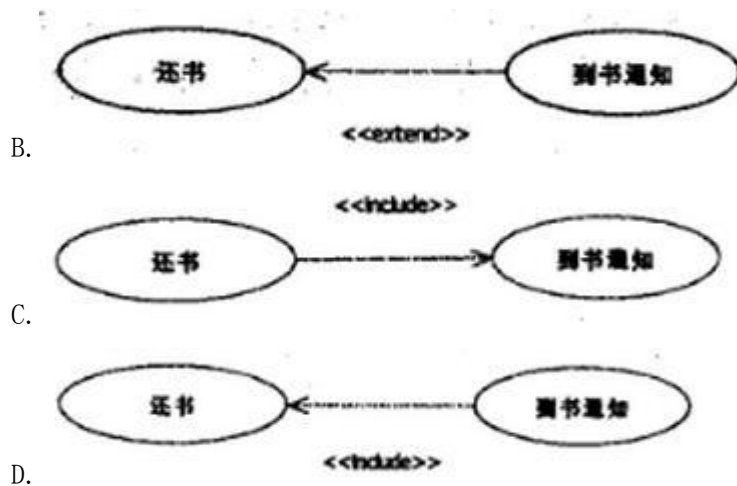
以下关于用例图的叙述中，不正确的是(44)。图书馆管理系统需求中包含“还书”用例和“到书通知”用例，对于“还书”用例，应先查询该书是否有人预定，若有则执行“到书通知”。“还书”用例和“到书通知”用例是(45)关系，以下用例图中，(46)是正确的。管理员处理“还书”用例时，需要先执行“验证身份”用例，那么“还书”用例和“验证身份”用例之间是(47)关系。

- (44)A. 系统用例图反映了整个系统提供的外部可见服务  
B. 系统用例图对系统的协作建模  
C. 用例图主要包含用例、参与者及其之间关系三个要素  
D. 系统用例图对系统的需求建模

- (45)A. 关联                      B. 扩展                      C. 包含                      D. 泛化



- (46)A.



(47) A. 关联

B. 扩展

C. 包含

D. 泛化

**【答案】** B B B C

**【解析】**

用例图展现了一组用例、参与者以及它们之间的关系；通常包括：用例；参与者；扩展关系、包含关系。

用例是对一组动作序列的描述，系统执行这些动作将产生一个对特定的参与者有价值而且可观察的结果。

用例图用于对系统的静态用例视图进行建模。这个视图主要支持系统的行为，即该系统在它的周边环境的语境中提供的外部可见服务。

当对系统的静态用例视图建模时，可以用下列两种方式来使用用例图。

对系统的语境建模。对一个系统的语境进行建模，包括围绕整个系统画一条线，并声明有哪些参与者位于系统之外并与系统进行交互。在这里，用例图说明了参与者以及他们所扮演的角色的含义。

对系统的需求建模。对一个系统的需求进行建模，包括说明这个系统应该做什么（从系统外部的一个视点出发），而不是考虑系统应该怎么做。在这里，用例图说明了系统想要的行为。通过这种方式，用例图使我们能够把整个系统看作一个黑盒子。可以观察到系统外部有什么，系统怎样与哪些外部事物相互作用，但却看不到系统内部是如何工作的。

(45、46、47) 用例的委托扩展两种方式：

包含关系：使用包含(include)用例来封装一组跨越多个用例的相似动作(行为片断)，以便多个基(Base)用例复用，做基用例的时候，必然会做它所包含的事件。

扩展关系：将基用例中一段相对独立并且可选的动作，用扩展(Extension)用例加以封装，再让它从基用例中声明的扩展点(Extension Point)上进行扩展，从而使基用例行



为更简练和目标更集中，做基事件之后，我可能做扩展事件，也可能不做。

用面向对象方法设计了一个父类 File 和两个子类 DiskFile 和 TapeFile，这两个子类继承了其父类的 open 方法，并给出不同的实现。不同的子类执行 open 方法时，有不同的行为，这种机制称为(48)。

- (48) A. 继承                      B. 多态                      C. 消息传递                      D. 关联

**【答案】B**

**【解析】**

多态性：是多种表现形式；多态性的实现，一般通过在派生类中重定义基类的虚函数来实现。

本题中给定一个方法，不同的子类行为不同，这是多态机制。

在计算机系统中，系统的(49)可以用  $MTTF / (1 + MTTF)$  来度量，其中 MTTF 为平均无故障时间。

- (49) A. 可靠性                      B. 可用性                      C. 可维护性                      D. 健壮性

**【答案】A**

**【解析】**

MTTF 是用来描述可靠性的指标。

修改现有软件系统的设计文档和代码以增强可读性，这种行为属于(50)维护。

- (50) A. 正确性                      B. 适应性                      C. 完善性                      D. 预防性

**【答案】D**

**【解析】**

改正性维护：是指在使用过程中发现了隐蔽的错误后，为了诊断和改正这些隐蔽错误而修改软件的活动

适应性维护：是指为了适用变化了的环境而修改软件的活动

完善性维护：是指为了扩充或完善原有软件的功能或性能而修改软件的活动

预防性维护：是指为了提高软件的可维护性和可靠性、为未来的进一步改进打下基础而修改软件的活动

题干中修改文档和代码提高可读性，提高可读性利于提高可维护性，所以应该属于预防

性维护。

以下不属于系统测试范畴的是\_(51)。

- (51) A. 单元测试                      B. 安全测试                      C. 强度测试                      D. 性能测试

**【答案】A**

**【解析】**

单元测试是对程序模块进行的测试，不属于系统测试的范畴。

以下关于文档测试的说法中，不正确的是\_(52)。

- (52) A. 文档测试需要仔细阅读文档，检查每个图形  
B. 文档测试需要检查文档内容是否正确和完善  
C. 文档测试需要检查标记是否正确性  
D. 文档测试需要确保大部分示例经过测试

**【答案】D**

**【解析】**

文档中的示例应像用户一样载入和使用样例。如果是一段程序，就输入数据并执行它以每一个模板制作文件，确认它们的正确性。

软件测试的对象不包括\_(53)。

- (53) A. 软件代码                      B. 软件文档                      C. 质量保证方法                      D. 相关数据

**【答案】C**

**【解析】**

软件测试的对象为：程序、数据和文档。

测试用例的三要素不包括\_(54)。

- (54) A. 输入                      B. 预期输出                      C. 执行条件                      D. 实际输出

**【答案】D**

**【解析】**

IEEE 829 标准中列出测试用例应该包含的重要信息：标识符、测试项、输入说明、输出说明、环境要求、特殊过程要求、用例之间的依赖性。结合本题来看应该选择选项 D。

以下关于软件测试原则的叙述中，正确的是(55)。

- ①测试开始得越早，越有利于发现缺陷
- ②测试覆盖率和测试用例数量成正比
- ③测试用例既需选用合理的输入数据，又需要选择不合理的输入数据
- ④应制定测试计划并严格执行，排除随意性
- ⑤采用合适的测试方法，可以做到穷举测试
- ⑥程序员应尽量测试自己的程序

(55) A. ①②③④⑤⑥      B. ①②③④⑤      C. ①②③④      D. ①③④

**【答案】D**

**【解析】**

软件测试原则：所有的测试都应追溯到用户需求；应尽早并不断地进行测试；测试工作应避免由原开发软件的人或小组来承担（单元测试除外）；穷举测试是不可能的，测试需要终止；充分重视测试中的群集现象；严格按照测试计划来进行，避免随意性。

以下关于测试时机的叙述中，正确的是(56)。

- ①应该尽可能早地进行测试
- ②软件中的错误暴露得越迟，则修复和改正错误所花费的代价就越高
- ③应该在代码编写完成后开始测试
- ④项目需求分析和设计阶段不需要测试人员参与

(56) A. ①②③④      B. ①②③      C. ①②      D. ①

**【答案】C**

**【解析】**

③④违背了测试应尽早开始的原则。

以下属于软件测试工具的是(57)。

- ①JTest    ②LoadRunner
- ③Visual Studio    ④JBuilder

(57) A. ①②③④      B. ①②③      C. ①②      D. ①

**【答案】C**

**【解析】**

①Java 单元测试工具 JTest；②是一种预测系统行为和性能的负载测试工具；③④开发相关的工具

兼容性测试不包括 (58)。

- |                |              |
|----------------|--------------|
| (58)A. 软件兼容性测试 | B. 硬件兼容性测试   |
| C. 数据兼容性测试     | D. 操作人员兼容性测试 |

**【答案】D**

**【解析】**

兼容性测试包括有软件、硬件、数据、平台等兼容性测试。

根据输出对输入的依赖关系设计测试用例的黑盒测试方法是 (59)。

- |               |         |           |        |
|---------------|---------|-----------|--------|
| (59)A. 等价类划分法 | B. 因果图法 | C. 边界值分析法 | D. 场景法 |
|---------------|---------|-----------|--------|

**【答案】B**

**【解析】**

因果图法：是从自然语言书写的程序规格说明的描述中找出因（输入条件）和果（输出或程序状态的改变），通过因果图转换为判断表。

以下关于边界值测试法的叙述中，不正确的是 (60)。

- (60)A. 边界值分析法仅需考虑输入域边界，不用考虑输出域边界
- B. 边界值分析法是对等价类划分方法的补充
- C. 错误更容易发生在输入输出边界上而不是输入输出范围的内部
- D. 测试数据应尽可能选取边界上的值

**【答案】A**

**【解析】**

边界值分析法需要考虑输入域的边界和输出域的边界。

一个程序的控制流图中有 6 个节点，10 条边，在测试用例数最少的情况下，确保程序中每个可执行语句至少执行一次所需要的测试用例数的上限是 (61)。

- |          |      |      |      |
|----------|------|------|------|
| (61)A. 2 | B. 4 | C. 6 | D. 8 |
|----------|------|------|------|

【答案】C

【解析】

$V(g) = \text{边} - \text{节点} + 1 = 10 - 6 + 2 = 6$ ;  $V(g)$  表示实现基本路径覆盖测试用例的最大数量。

对于逻辑表达式  $((b1 \& b2) || in)$ ，需要 (62) 个测试用例才能完成条件组合覆盖。

(62) A. 2

B. 4

C. 8

D. 26

【答案】B

【解析】

多条件覆盖 (MCC): 也称条件组合覆盖, 设计足够的测试用例, 使得使得每个判定中条件的各种可能组合都至少出现一次

本题中  $\&$  被认为是位运算, 则 MCC 的测试用例数应为:  $2^2=4$

如果作为逻辑与运算, 则有 3 个条件, 符合 MCC 的测试用例数应为:  $2^3=8$

测试执行过程的阶段不包括 (63)。

(63) A. 初测期

B. 系统测试期

C. 细测期

D. 回归测试期

【答案】B

【解析】

测试执行过程的阶段为: 初测期、细测期、和回归测试期。

初测期——测试主要功能和关键的执行路径, 排除主要障碍。

细测期——依据测试计划和测试用例, 逐一测试大大小小的功能、方方面面的特性、性能、用户界面、兼容性、可用性等等; 预期可发现大量不同性质、不同严重程度的错误和问题。

回归测试期——系统已达到稳定, 在一轮测试中发现的错误已十分有限; 复查已知错误的纠正情况, 未引发任何新的错误时, 终结回归测试。

以下关于回归测试的叙述中, 不正确的是 (64)。

(64) A. 回归测试是为了确保改动不会带来不可预料的后果或错误

B. 回归测试需要针对修改过的软件成分进行测试

C. 回归测试需要能够测试软件的所有功能的代表性测试用例

D. 回归测试不容易实现自动化

**【答案】A**

**【解析】**

回归测试是指修改了旧代码后，重新进行测试以确认修改没有引入新的错误或导致其他代码产生错误。

以下属于测试停止依据的是 (65)。

- ①测试用例全部执行结束 ②测试覆盖率达到要求  
③测试超出了预定时间 ④查出了预定数目的故障  
⑤执行了预定的测试方案 ⑥测试时间不足

(65) A. ①②③④⑤⑥      B. ①②③④⑤      C. ①②③④      D. ①②③

**【答案】B**

**【解析】**

⑥测试时间不足不应作为测试停止的依据

以下关于测试方法的叙述中，不正确的是 (66)。

- (66) A. 根据被测代码是否可见分为白盒测试和黑盒测试  
B. 黑盒测试一般用来确认软件功能的正确性和可操作性  
C. 静态测试主要是对软件的编程格式 M 结构等方面进行评估  
D. 动态测试不需要实际执行程序

**【答案】D**

**【解析】**

动态测试是指通过人工或使用工具运行程序进行检查、分析程序的执行状态和程序的外部表现。所以选项 D 是错误的描述。

以下关于性能测试的叙述中，不正确的是 (67)。

- (67) A. 性能测试的目的是为了验证软件系统是否能够达到用户提出的性能指标  
B. 性能测试不用于发现软件系统中存在的性能瓶颈  
C. 性能测试类型包括负载测试，强度测试，容量测试等  
D. 性能测试常通过工具来模拟大量用户操作，增加系统负载

**【答案】B**

**【解析】**

性能测试用来保证产品发布后系统的性能能够满足用户需求。性能测试通常存在性能调优与性能评测两种性能测试策略。

性能评测主要包括：在真实环境下，检查系统服务等级的满足情况，评估并报告整个系统的性能；对系统的未来容量作出预测和规划。

性能评测是性能调优的基础，性能调优的步骤如下：查找形成系统瓶颈或者故障的根本原因；进行性能调整和优化；评估性能调整的效果。

不同加密机制或算法的用途、强度是不相同的，一个软件或系统中的加密机制使用是否合理，强度是否满足当前要求，需要通过测试来完成，通常 (68) 是测试的一个重要手段。

(68) A. 模拟加密      B. 模拟解密      C. 漏洞扫描      D. 算法强度理论分析

**【答案】B**

**【解析】**

加密机制是保护数据安全的重要手段，加密的基本过程就是对原来为明文的文件或数据，按某种算法进行处理，使其成为不可读的密文。由于不同加密机制的用途及强度不同，因此一个信息系统中加密机制使用是否合理，强度是否满足当前需要，需要通过测试来检验，通常模拟解密是测试的一个重要手段。

安全日志是软件产品的一种被动防范措施，是系统重要的安全功能，因此安全日志测试是软件系统安全性测试的重要内容，下列不属于安全日志测试基本测试内容的是 (69)。

(69) A. 对安全日志的完整性进行测试，测试安全日志中是否记录包括用户登录名称、时间、地址、数据操作行为以及退出时间等全部内容

B. 对安全日志的正确性进行测试，测试安全日志中记录的用户登录、数据操作等日志信息是否正确

C. 对日志信息的保密性进行测试，测试安全日志中的日志信息是否加密存储，加密强度是否充分

D. 对于大型应用软件系统，测试系统是否提供安全日志的统计分析能力

**【答案】C**

**【解析】**

日志应当记录所有用户访问系统的操作内容，包括登录用户名称、登录时间、浏览数据

动作、修改数据动作、删除数据动作、退出时间、登录机器的 IP 等。

安全日志测试：根据业主要求或设计需求，对日志的完整性、正确性进行测试；测试安全日志是否包含了日志内容的全部项目，是否正确；对于大型应用软件，系统是否提供了安全日志的智能统计分析能；是否可以按照各种特征项进行日志统计，分析潜在的安全隐患，及时发现非法行为。

下到关于 DoS 攻击的描述中，错误的是 (70)。

(70) A. oS 攻击通常通过抑制所有或流向某一特定目的端的消息，从而使系统某一实体不能执行其正常功能，产生服务拒绝

B. DoS 攻击不需入目标系统，仅从外部就可实现攻击

C. 只要软件系统内部没有漏洞，DoS 攻击就不可能成功

D. 死亡之 Ping、Land 攻击、UDP 洪水、Smurf 攻击均是常见的 DoS 攻击手段

**【答案】C**

**【解析】**

服务拒绝：当一个实体不能执行它的正常功能，或它的动作妨碍了别的实体执行它们的正常功能的时候，便发生服务拒绝。这种攻击不一定需要目标系统存在漏洞，如 UDP 洪水。

Software entities are more complex for their size than perhaps any other human construct, because no two parts are alike (at least above the statement level). If they are, we make the two similar parts into one, a (71), open or closed. In this respect software systems differ profoundly from computers, buildings, or automobiles, where repeated elements abound.

Digital computers are themselves more complex than most things people build; they have very large numbers of states. This makes conceiving, describing, and testing them hard. Software systems have orders of magnitude more (72) than computers do. Likewise, a scaling-up of a software entity is not merely a repetition of the same elements in larger size; it is necessarily an increase in the number of different elements. In most cases, the elements interact with each other in some (73) fashion, and the complexity of the whole increases much more than linearly.

The complexity of software is a(an) (74) property, not an accidental one. Hence



descriptions of a software entity that abstract away its complexity often abstract away its essence. Mathematics and the physical sciences made great strides for three centuries by constructing simplified models of complex phenomena, deriving, properties from the models, and verifying those properties experimentally. This worked because the complexities (75) in the models were not the essential properties of the phenomena. It does not work when the complexities are the essence.

Many of the classical problems of developing software products derive from this essential complexity and its nonlinear increases with size. Not only technical problems but management problems as well come from the complexity.

- |                |              |               |                |
|----------------|--------------|---------------|----------------|
| (71)A. task    | B. job       | C. subroutine | D. program     |
| (72)A. states  | B. parts     | C. conditions | D. expressions |
| (73)A. linear  | B. nonlinear | C. parallel   | D. additive    |
| (74)A. surface | B. outside   | C. exterior   | D. essential   |
| (75)A. fixed   | B. included  | C. ignored    | D. tabilized   |

**【答案】** C   A   B   D   C

**【解析】**

软件实体规模上或许比其他任何人类创造的结构要更为复杂,因为没有两个部分是一样的(至少在语句级以上)。如果他们一样,我们便将这两个相似的部分合到一起,成为一个子程序,打开或关闭。在这一点上,软件系统与计算机、建筑或汽车经常混合使用一些重复的元件有很大区别。

数字计算机本身比人类所建造的大多数事物都要复杂,他们有超级多的状态。这使得对他们进行想象、描述和测试都很困难。软件系统的状态数目更是比计算机的超出几个数量级。同理,软件实体的扩展不单是这一批元件变大一点,它必将是大量不同元件都有增加。在大多数情况下,构件以非线性的方式相互作用,而整体的复杂性远超线性增加。

软件的复杂的特点具有必然性,并非偶然。因此,对于软件实体的描述,剥离了它的复杂性往往就等于剥离了它的本质。过去这三个世纪,通过对复杂的现象构建出简化模型,从模型的属性再倒推,并通过实验验证这些属性,数学和物理科学取得了长足的进步。这之所以行之有效,是因为模型中忽略掉的复杂特性并不是现象中重要的本质属性。而当这些复杂性很重要时,这种方法就会失效。



## 试题一

阅读下列 C 程序，回答问题 1 至问题 3，将解答填入答题纸的对应栏内。

### 【C 程序】

```
int count(int x,int z){  
    int y=0;  
    while(x>0){ //1  
        if(x==1) //2  
            y=7; //3  
        else{ //4  
            y=x+z+4;  
            if (y==7||y==21) //5, 6  
                x=1; //7  
        }  
        x--; //8  
    }  
    return y; //9  
}
```

### 【问题 1】(3 分)

请针对上述 C 程序给出满足 100%DC（判定覆盖）所需的逻辑条件。

$x>0$ ;  $x\leq 0$

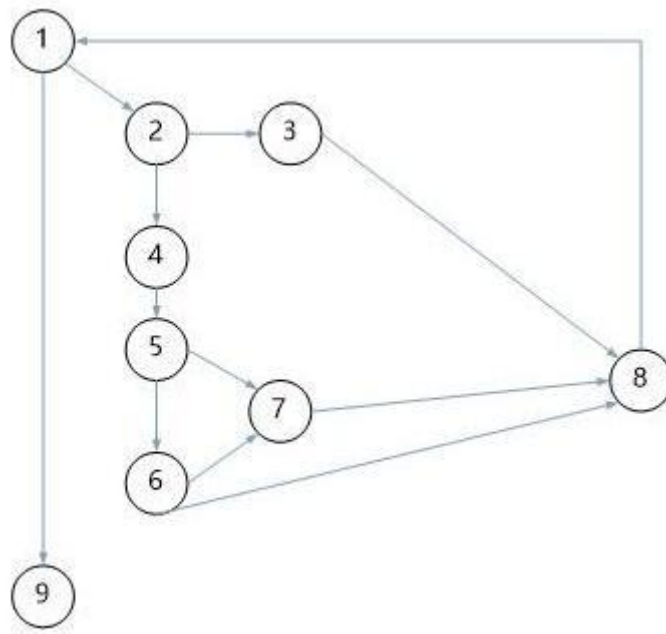
$x==1$ ;  $x!=1$

$y==7$  或者  $y==21$ ;  $y!=7$  且  $y!=21$

解析：本题中的判定有  $x>0$ ;  $x==1$ ;  $y==7||y==21$ ；三个判定点，所以要符合 100%判定覆盖的要求，就需要使得每个判定结果的真假值都出现 1 次，即： $x>0$ ;  $x\leq 0$ ;  $x==1$ ;  $x!=1$ ;  
 $y==7$  或者  $y==21$ ;  $y!=7$  且  $y!=21$

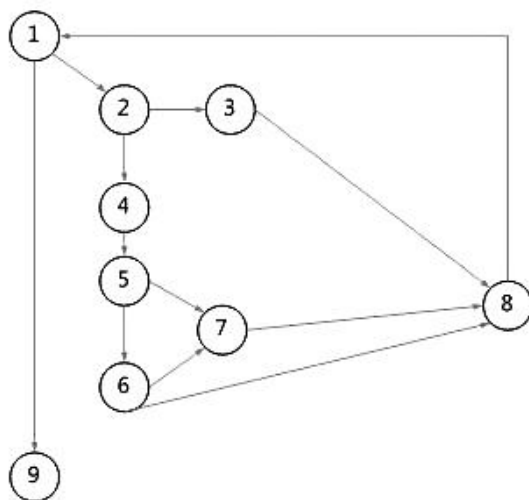
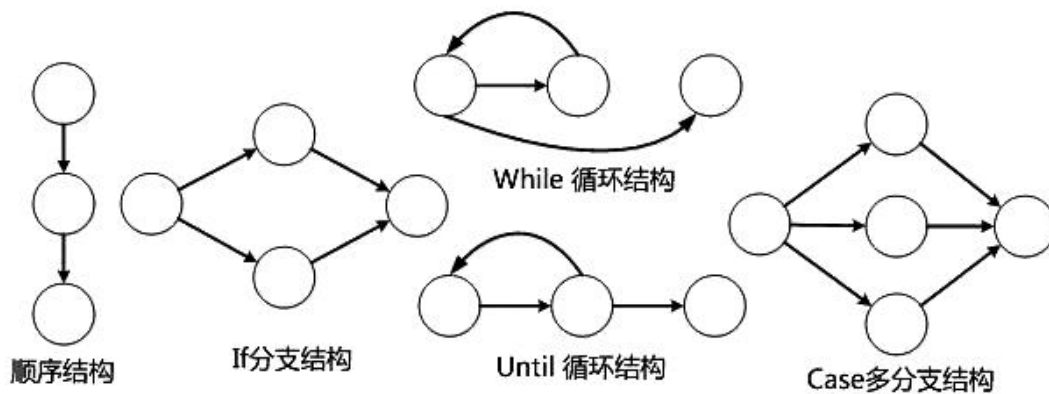
### 【问题 2】(7 分)

请画出上述程序的控制流图，并计算其控制流图的环路复杂度  $V(G)$ 。



$$V(g)=5$$

解析：控制流图是描述程序控制流的一种图示方法。其基本符号有圆圈和箭线：圆圈为控制流图中的一个结点，表示一个或多个无分支的语句；带箭头的线段称为边或连接，表示控制流。基本结构如下所示：



控制流程图的环路复杂性  $V(G)$  等于：

- (1) 控制流程图中的区域个数。
- (2) 边数-结点数+2。
- (3) 判定数+1。

$V(g)=5$

### 【问题 3】(5 分)

请给出问题 2 中控制流图的线性无关路径。

线性无关路径是指包括一组以前没有处理的语句或条件的一条路径。从控制流图来看，一条线性无关路径是至少包含有一条在其他线性无关路径中从未有过的边的路径：

1: 1, 9

2: 1, 2, 3, 8, 1, 9

3: 1, 2, 4, 5, 7, 8, 1, 9

4: 1, 2, 4, 5, 6, 7, 8, 1, 9

5: 1, 2, 4, 5, 6, 8, 1, 9

试题二

阅读下列说明，回答问题 1 和问题 2，将解答填入答题纸的对应栏内。

【说明】

某航空公司的会员卡分为普卡、银卡、金卡和白金卡 4 个级别，会员每次搭乘该航空公司航班均可能获得积分，积分规则如表 2-1 所示。此外，银卡及以上级别会员有额外积分奖励，奖励规则如表 2-2 所示。

舱位	舱位代码	积分
头等舱	F	200%*K
	Z	150%*K
	A	125%*K+
公务舱	C	150%*K
	D/I	125%*K
	R	100%*K
经济舱	Y	125%*K
	B/H/K/L/m/V	100%*K
	Q/X/U/E	50%*k
	P/S/G/O/JN[FW	O

表 2-2 额外积分奖励规则

会员级别	普卡	银卡	金卡	白金卡
级别代码	F	S	G	p
额外积分奖励	0%	10%	25%	50%

公司开发了一个程序来计算会员每次搭乘航班历累积的积分，程序的输入包括会员级别

B、舱位代码 C 和飞行公里数 K，程序的输出为本次积分 S 。其中，B 和 C 三字母且大小写不敏感，K 为正整数，S 为整数（小数部分四舍五入）。

【问题 1】（7 分）

采用等价类型划分法对该程序进行测试，等价类表如 2-3 所示，请补充空（1）-（7）

表 2-3 等价类表

输入条件	有效等价类	编号	无效等价类	编号
会员等级 B	F	1	非字母	12
	S	2	非单个字母	13
	G	3	(5)	14
	(1)	4		
舱位代码 C	F	5	非字母	15
	(2)	6	(6)	16
	(3)	7		
	R/B/H/K/L/M/W	8		
	Q/X/U/E	9		
	P/S/G/O/I/V/N/T	10		
飞行公里数 K	(4)	11	非整数	17
			(7)	18

1： P 2： Z/C 3： A/D/I /Y 4： 正整数

5： 非 FSGP 的字母 6： 非规定仓位的字母 7： 非正数

解析：等价类划分原则：

在输入条件规定了取值范围或值的个数的情况下，可以确定一个有效等价类和两个无效等价类

在输入条件规定了输入值得集合或者规定了“必须如何”的条件的情况下，可以确立一个有效等价类和一个无效等价类

在输入条件是一个布尔量的情况下，可确定一个有效等价类和一个无效等价类

在规定了输入数据的一组值（假定 n 个），并且程序要对每一个输入值分别处理的情况下，可确定 n 个有效等价类和一个无效等价类

在规定了输入数据必须遵守的规则的情况下，可确定一个有效等价类（符合规则）和若干个无效等价类（从不同角度违反规则）

在确知已划分的等价类中，各元素在程序处理中的方式不同的情况下，则应再将该等价类进一步地划分为更小的等价类

本题中会员等级的有效等价类应为：表格中指定的类：FSGP，由于每个等级是单独处理，



所以有 4 个等价类，无效等价类：应有非 FSGP 之内的字母，非字母，非单个字母等。结合起来所以 1 为 P；5 为非 FSGP 的字母；对于飞行公理 K，应是一个正整数，所以有效等价类为正整数，无效等价类为非整数，和非正整数；因此：4 为正整数；7 为非正整数。对于仓位代码 C，结合表格来看，R 和 B/H/K/L/M/W 放在一个有效等价类中，说明，出题的意思是仓位代码按照不同积分的程度分类（0，50%，100%，125%，150%，200%），即六个有效等价类，无效等价类应为：非字母和不在列表之内的字母。因此：2 为 Z/C 3 为 A/D/I/Y 6 为未按规定仓位的字母。

**【问题 2】（13 分）**

根据以上等价类表设计的测试用例如表 24 所示, 请补充空(1)~(13)。

表 2-4

编号	输入			覆盖等价类 (编号)	预期输出 S
	B	C	K		
1	F	F	500	1,5,11	(1)
2	S	Z	(2)	2,6,11	825
3	G	A	500	(3)	781
4	P	(4)	500	4,8,11	750
5	(5)	Q	500	1,9,11	250
6	F	P	500	1,10,11	(6)
7	(7)	P	500	12,10,11	N/A
8	(8)	F	500	13,5,11	N/A
9	A	Z	500	14,6,11	N/A
10	S	(9)	500	2,15,11	N/A
11	S	(10)	500	2,16,11	N/A
12	S	Q	(11)	2,9,17	(12)
13	S	P	(13)	2,10,18	N/A

按照相关的规则进行计算:

1: 1000 2: 500 3: 3, 7, 11 4: R/B/H/K/L/M/W

(有效等价类, 一个测试用例可以覆盖多个有效类)

5: F 6: 0 7: 9 8: GG

[illegible]

(无有效等价类，一个测试用例只能覆盖 1 个有效类)

### 试题三

阅读下列说明，回答问题 1 至问题 4，将解答填入答题纸的对应栏内。

#### 【说明】

某证券交易所为了方便提供证券交易服务，欲开发一个基于 Web 的证券交易平台。其主要功能包括客户开户，记录查询、存取款、股票交易等。客户信息包括姓名、Email（必填且唯一）、地址等；股票交易信息包括股票代码（6 位数字编码的字符串）、交易数量（100 的整数倍）、买/卖价格（单位：元，精确到分）。

系统要支持：

(1) 在特定时朝内 3000 个用户并发时，主要功能的处理能力至少要达到 128 个请求/秒，平均数据量 2KB/请求；

(2) 页面中采用表单实现客户信息、交易信息等的提交与交互，系统前端采用 HTML5 实现。

#### 【问题 1】（4 分）

在对此平台进行非功能测试时，需要测试哪些方面？

性能测试、安全性测试、兼容性测试、易用性测试

#### 【问题 2】（5 分）

在满足系统要支持的(1)时，计算系统的通信吞吐量。

$3000 \times 128 \times 2KB = 750MB$

#### 【问题 3】

每个字段的验证

字段的缺省值

表单中的输入

#### 【问题 4】（8 分）

(1) 针对股票代码：111111、数量：10 万、当前价格：6.00，设计 4 个股票交易的测试输入。

(2) 设计 2 个客户开户的测试输入，以测试是否存在 XSS、SQL 注入。

1: 111111, 10 万, 6（正确输入）

2: 1222, 10 万, 6（代码错误）

3: 111111, 0 万, 6 (数量错误)

4: 111111, 10 万, 0 (价格错误)

21: 姓名: 张三 or 1=1-- Email: q123@q123.com 地址: xxxxxxx

22: 姓名: 张三 Email: q123@q123.com 地址: xxxxxxx<script>alert('测试信息')</script>

解析: (1) 本题中有 3 条件, 应该设计一个测试用例覆盖 3 个有效等价类; 然后针对每个条件的无效等价类各设计一个测试用例. (2) 针对客户信息包括姓名、Email (必填且唯一)、地址, 等输入, 任意挑选一个输入文本框, 在保障 SQL 能运行的前提下, 输入带有注入式攻击和 XSS 攻击的特征内容即可。

#### 试题四

阅读下列说明，回答问题 1 至问题 5，将解答写在答题纸的对应栏内。

##### 【说明】

图 4-1 是银行卡应用的部分类图，图中属性和操作前的“+”和“-”分别表示公有成员和私有成员。银行卡 Account 有两种类型，借记卡 SavingAccount 和信用卡 CreditAccount。

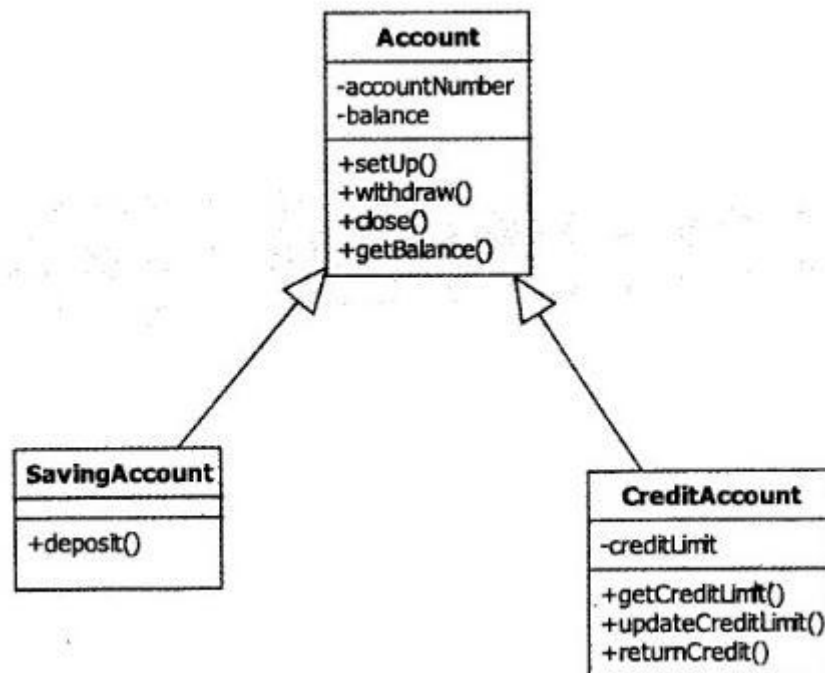


图 4-1 银行卡应用的部分类图

(1) 借记卡和信用卡都有卡号 `account Number` 和余额 `balance` 两个属性。借记卡的余额是正常余额，信用卡的余额是目前未还的金额，如果没有未还的金额，则为 0；有开户 `set Up`、取款 `withdraw`、查询余额 `get Balance` 和销户 `close` 四个方法。借记卡取钱时，要求取钱金额不能超过余额；而信用卡取钱金额不能超过信用额度，因此需要在子类中实现该方法。

(2) 借记卡可以存钱 `deposit`。

(3) 信用卡有信用额度 `creditLimit` 属性，可以查询信用额度 `getCreditLimit`、修改信用额度 `updateCreditLimit` 和还款 `returnCredit`。

现拟采用面向对象的方法进行测试。

##### 【问题 1】(3 分)

面向对象单元测试的主要对象是什么？

是针对程序的函数、过程或完成某以特定功能的程序块。

**【问题 2】（4 分）**

在继承关系上，若某方法在测试父类时已经测试过，那么在什么情况下在子类中也需要测试？

1：继承类的成员函数在子类中做了改动

2：成员函数调用了改动过的成员函数的部分

**【问题 3】（4 分）**

要测试方法 `deposit()` 时，还需要调用什么方法？给出测试序列。

`getBalance()`

先测试 `getBalance()` 再测试 `deposit()`

**【问题 4】（6 分）**

方法 `withdraw` 在基类 `Account` 中定义，但在两个子类中有不同的实现。这是面向对象的什么机制？这种情况在测试时如何进行？

多态

只需要在原有的测试分析和基础上增加对测试用例中输入数据的类型的考虑；先测试基类，然后再分别依据输入数据设计不同的测试用例。

**【问题 5】（3 分）**

给出类 `SavingAccount` 的最小测试序列。

先测试 `Account` 类，然后测试 `SavingAccount` 类

## 试题五

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】

某嵌入式控制软件通过采集传感器数值来计算输出控制率，为了提高数据采集的可靠性，使用三余度采集方法进行三个通道的数据采集。

1. 三余度通道数据采集及处理要求：

(1) 三通道采集值  $In\_U[0]$ 、 $In\_U[1]$ 、 $In\_U[2]$  的正常范围为  $[-3.0, 3.0]V$ ，且任意两通道间差值不大于  $0.5V$ ；

(2) 如果某通道采集值超过正常范围或者因为通道采集值造成与其它通道差值大于  $0.5V$ ，则该通道数据不满足要求；

(3) 如果三通道值均满足要求，则取三通道中差值较小的两通道数据的平均值；

(4) 如果仅有一个通道数据不满足采集要求，取满足要求的两个通道数据的平均值；

(5) 如果多于一个通道数据不满足采集要求，取安全值  $0V$ 。

2. 对采集数值计算控制率的具体处理算法如下：

(1) 如果依据采集数据计算的控制率  $C1$  与目前实际控制率  $C0$  差值不大于  $0.01$  则使用本周期计算控制率  $C1$  进行输出控制，否则使用目前实际控制率  $C0$  输出控制。不上报传感器故障；

(2) 如果连续 3 个周期计算的控制率  $C1$  与目前实际控制率  $C0$  差值大于  $0.01$ ，则上报传感器三级故障，连续超差计数清零，使用目前实际控制率  $C0$  输出控制；如果已经连续 3 个周期控制率超差，并上报三级故障，但第 4 个周期计算的控制率  $C1$  与目前实际控制率  $C0$  差值不大于  $0.01$ ，则清除三级故障上报；

(3) 如果累计大于等于 10 个周期计算的控制率  $C1$  与目前实际控制率  $C0$  差值大于  $0.01$ ，则上报传感器二级故障，使用目前实际控制率输出控制；

(4) 如果累计大于等于 100 个周期计算的控制率  $C1$  与目前实际控制率  $C0$  差大于  $0.01$ ，则上报传感器一级故障，并清除二级故障，并切断输出控制（输出安全值 0）

(5) 如果低级故障和高级故障同时发生，则按高级故障处理。

### 【问题 1】（9 分）

为了测试采集算法，在不考虑测量误差的情况下，设计了表 5-1 所示的测试用例，请填写该表中的空 (1)~(6)。

表 5-1 测试用例

序号	输 入			输出 Out_A1
	In_U[0]	In_U[1]	In_U[2]	预期输出（保留两位小数）
1	0.0V	0.0V	0.0V	0.00V
2	2.0V	2.3V	1.8V	(1)
3	1.5V	1.6V	1.3V	(2)
4	2.8V	2.6V	2.0V	(3)
5	-3.0V	-3.1V	-2.8V	(4)
6	2.0V	1.4V	2.6V	(5)
7	3.1V	2.8V	3.2V	(6)

1: 1.9 2: 1.55 3: 2.7

4: -2.90 5: 0 6: 0

解析：（1）信道之差不大于 0.5，所以三个信道都满足要求，按规则（3）处理，即取 1.8 和 2.0 的平均值

（2）信道之差不大于 0.5，所以三个信道都满足要求，按规则（3）处理，即取 1.5 和 1.6 的平均值

（3）信道之差 2.8 与 2.0 直接大于 0.5，2.6 与 2.0 之间大于 0.5，所以 2.0 不符合要求，按规则（4）处理，即取 2.6 和 2.8 的平均值

（4）信道之差不大于 0.5，-3.1 超过【-3.0，3.0】的范围，不满足要求，按规则（4）处理，即取-3.0 和-2.8 的平均值

（5）由于任意两个信道之间的差值都超过 0.5，所以所有信道都不符合要求，按规则（5）处理，即取值为 0

（6）由于 3.1 和 3.2 都超过【-3.0，3.0】的范围，按规则（5）处理，即取值为 0

### 【问题 2】（9 分）

为了测试控制率计算算法，在不考虑测量误差的情况下，设计了表 5-2 所示的用例，请完善其中的空（1）～（6）。

表 5-2 测试用例

序号	前置条件		输 入		输出（预期结果）	
	控制率超差 连续计数	控制率超差 累计计数	计算控制率 $C_1$	实际控制率 $C_0$	输出控制率	上报故障
1	0	0	1.632	1.638	1.632	无
2	0	0	1.465	1.454	(1)	无
3	(2)	6	2.358	2.369	2.369	三级故障
4	1	(3)	1.569	1.557	1.557	二级故障
5	2	9	2.221	2.234	2.234	(4)
6	0	99	1.835	1.822	(5)	一级故障
7	2	99	2.346	2.357	0	(6)

1: 1.454 2: 2 3: 9

4: 二级故障 5: 0 6: 一级故障

解析: (1) 按照规则 1,  $1.465 - 1.454 = 0.011$  大于 0.01, 所以应该为实际控制率

(2) 是三级故障, 三级故障要求连续 3 个周期, 所以前置条件应该为 2

(3) 是二级故障, 二级故障要求累计 10 个周期, 所以前置条件应该为 9

(4) 前置累计次数已经达到 9, 在加上本次, 就连续次数达到 10, 所以应该为二级故障

(5) 是一级故障, 一级故障的输出值为 0

(6) 由于输出值为 0, 应属于一级故障

### 【问题 3】(2 分)

测试人员在设计测试用例进行采集算法测试时, 发现本项目的三余度采集值的具体处理算法存在 1 处缺陷, 请指出此处缺陷。

缺陷: 当三个通道的数据只差不超过 0.5, 而且两两之间的差值有两个相等时, 存在取值问题。如: 测试数据为: 2.0 &nbsp;    2.2 2.4 这时, 存在取值问题, 是取 2.1 还是 2.3?