

# 程序员

## 2018年上半年试题

本试卷为：样式2

样式1：适用于模拟考试，所有答案在最后面。

样式2：适用于复习，每道题的题目和答案在一起。

本试卷由**跨步软考**提供

我们目前提供的免费服务有：

- 手机APP刷题
- 网页版刷题
- 真题pdf版下载
- 视频课程下载
- 其他资料下载

更多免费服务请访问我们的官网：<http://kuabu.xyz>

你也可以关注我们的微信公众号：**跨步软考**

如果您发现试题有错误，您可以通过以下方式联系我们

- 客服邮箱：service@kuabu.xyz
- 备用邮箱：kuabu@outlook.com
- 您也可以在微信公众号后台留言

本文档所有权归**跨步软考**(kuabu.xyz)，您可以传播甚至修改本文档，但是必须标明出自“**跨步软考 (kuabu.xyz)**”

## 上午综合试卷

第1题：某编辑在编辑文稿时发现如下错误，其中最严重的错误是（ ）。

- A. 段落标题编号错误
- B. 将某地区名列入了国家名单
- C. 语句不通顺、有明显的错别字
- D. 标点符号、字体、字号不符合要求

参考答案：B

解析：

本题考查信息处理实务方面的基础知识。

文稿编辑时发生文字错误是常见的，难免的，但严重的错误影响很大。对一般的错误可以加强校对来尽量减少错误。但对严重的错误一定要引起领导人和工作人员的特别重视，在事先加强案例教育，事中特别注意校对，发生严重错误后便要严肃处理相关责任人。

国家名单，有可能引发政治问题。对部分公民的某种歧视，泄露本单位的商业秘密等也会产生严重问题。

第2题：某县有6.6万个贫困户，县委组织人员调查这6.6万个贫困户经济收入，从中抽取1800个贫困户的经济收入进行分析。请问本次调查的总体、个体、样本及样本容量分别为（ ）。

- A. 6.6万个贫困户经济收入、每个贫困户的经济收入、1800、1800个贫困户
- B. 6.6万个贫困户、1800个贫困户经济收入、每个贫困户的经济收入、1800
- C. 6.6万个贫困户、每个贫困户的经济收入、1800个贫困户经济收入、1800
- D. 6.6万个贫困户、每个贫困户的经济收入、1800、1800个贫困户经济收入

参考答案：C

解析：

本题考查信息处理方面的问题。

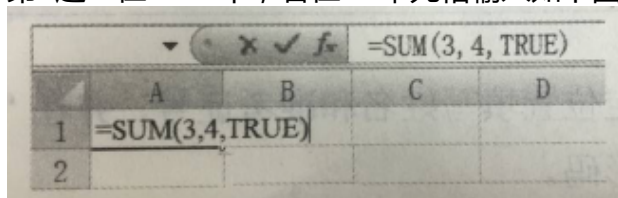
总体（母体）：调查对象的全体，在这里是6.6万个贫困户；

个体：总体中每一个考察对象，在这里就是每个贫困户的经济收入；

样本：总体中所抽取的一部分个体，在这里是1800个贫困户的经济收入；

样本容量：样本中个体的数目，因此是1800。

第3题：在Excel中，若在A1单元格输入如下图所示的内容，则A1的值为（ ）。



- A. 7
- B. 8
- C. TRUE
- D. #NAME?

参考答案：B

解析：

本题考查Excel的公式和函数。

在EXCEL里，逻辑值结果为True的时候在进行运算会转化为1，如果是False会转化为0。

SUM函数是进行求和，所以这里的答案是 $=3+4+1=8$ ，选择B。

第4题：在Excel中，单元格L3内容为“软件工程技术”，若要取单元格L3前两个字“软件”放入单元格M3中，则在M3中可输入（ ），并按下回车键即可。

- A. =LEFTB(M3,2)
- B. =LEFT(M3,2)
- C. =LEFTB(L3,2)
- D. =LEFT(L3,2)

参考答案：D

解析：

本题考查Excel的公式和函数。

LEFT函数：根据所指定的字符数，LEFT 返回文本字符串中第一个字符或前几个字符。

LEFTB函数：LEFTB 基于所指定的字节数返回文本字符串中的第一个或前几个字符。

根据两个函数的区别，可以知道如果需要获取L3的前两个字，需要用“=LEFT ( L3 , 2 )”或者“=LEFTB ( L3 , 4 )”。一个汉字在计算机里面占用2个字节。

第5题：电子邮件地址“zhangli@mail.ceiaec.org”中的zhangli、@和mail.ceiaec.org分别表示用户信箱的（ ）。

- A. 邮件接收服务器域名、帐号和分隔符

- B. 邮件接收服务器域名、分隔符和帐号
- C. 帐号、分隔符和邮件接收服务器域名
- D. 帐号、邮件接收服务器域名和分隔符

参考答案：C

解析：

一个电子邮件地址由三部分组成，第一部分代表用户信箱的帐号，第二部分是分隔符，第三部分是用户信箱的邮件接收服务器域名，用以标识其所在的位置。

一个电子邮件地址，例如zhangli@mail.ceiaec.org，由三部分组成，其中，第一部分“zhangli”代表用户信箱的帐号，对于同一个邮件接收服务器来说，这个帐号必须是唯一的；第二部分“@”是分隔符；第三部分“mail.ceiaec.org”是用户信箱的邮件接收服务器域名，用以标识其所在的位置。

第6题：程序计数器（PC）是用来指出下一条待执行指令地址的，它属于（ ）中的部件。

- A. CPU
- B. RAM
- C. Cache
- D. USB

参考答案：A

解析：

本题考查计算机系统基础知识。

CPU内部结构大概可以分为控制单元、运算单元、存储单元和时钟等几个主要部分。

运算器是计算机对数据进行加工处理的中心，它主要由算术逻辑部件（ALU：Arithmetic and Logic Unit）、寄存器组和状态寄存器组成。ALU主要完成对二进制信息的定点算术运算、逻辑运算和各种移位操作。通用寄存器组（典型代表是累加寄存器）来保存参加运算的操作数和运算的中间结果。状态寄存器在不同的机器中有不同的规定，程序中，状态位通常作为转移指令的判断条件。

控制器是计算机的控制中心，它决定了计算机运行过程的自动化。它不仅要保证程序的正确执行，而且要能够处理异常事件。主要有程序计数器PC、指令译码器、指令寄存器IR和操作控制器组成。

其中PC又称为“指令计数器”，它保存了下一条要执行指令的地址，由于大多数指令的地址与前指令的修改的过程通常只是简单的将PC加1。如果遇到转移指定的时候，直接修改为转移指令给出的地址。

指令寄存器用于储存现在正在被运行的指令。一条指令包括操作码和操作数，操作码就是表明了此指令要干什么，而这便是指令译码器的作用。

时序控制逻辑要为每条指令按时间顺序提供应有的控制信号。

第7题：以下关于主流固态硬盘的叙述中，正确的是（ ）。

- A. 存储介质是磁表面存储器，比机械硬盘功耗高
- B. 存储介质是磁表面存储器，比机械硬盘功耗低
- C. 存储介质是闪存芯片，比机械硬盘功耗高
- D. 存储介质是闪存芯片，比机械硬盘功耗低

参考答案：D

解析：

固态硬盘（Solid State Drives），简称固盘，固态硬盘（Solid State Drive）用固态电子存储芯片阵列而制成的硬盘，由控制单元和存储单元（FLASH芯片、DRAM芯片）组成。

低功耗：固态硬盘的功耗上要低于传统硬盘。

固态硬盘与机械硬盘相比具有读写速度快、防震抗摔性、低功耗、无噪音、工作温度范围大、轻便等特点。

第8题：CPU中可用来暂存运算结果的是（ ）。

- A. 算逻运算单元
- B. 累加器
- C. 数据总线
- D. 状态寄存器

参考答案：B

解析：

本题考察计算机系统基础知识。

CPU内部结构大概可以分为控制单元、运算单元、存储单元和时钟等几个主要部分。

运算器是计算机对数据进行加工处理的中心，它主要由算术逻辑部件（ALU：Arithmetic and Logic Unit）、寄存器组和状态寄存器组成。

ALU主要完成对二进制信息的定点算术运算、逻辑运算和各种移位操作。

通用寄存器组(典型代表是累加寄存器)来保存参加运算的操作数和运算的中间结果。

状态寄存器在不同的机器中有不同的规定，程序中，状态位通常作为转移指令的判断条件。

控制器是计算机的控制中心，它决定了计算机运行过程的自动化。它不仅要保证程序的正确执行，而且要能够处理异常事件。主要有程序计数器PC、指令译码器、指令寄存器IR和操作控制器组成。

其中PC又称为“指令计数器”，它保存了下一条要执行指令的地址，由于大多数指令的地址与前指令的修改的过程通常只是简单的将PC加1。如果遇到转移指定的时候，直接修改为转移指令给出的地址。

指令寄存器用于储存现在正在被运行的指令。一条指令包括操作码和操作数，操作码就是

表明了此指令要干什么, 而这便是指令译码器的作用。

时序控制逻辑要为每条指令按时间顺序提供应有的控制信号。

第9题: 微机系统中系统总线的( )是指单位时间内总线上传送的数据量。

- A. 主频
- B. 工作频率
- C. 位宽
- D. 带宽

参考答案: D

解析:

本题考察计算机的相关性能情况。

系统总线的性能指标:

带宽: 单位时间上传送的数据量, 即每秒钟传送的最大稳态数据传输率;

位宽: 能同时传送的二进制数据的位数, 或数据总线的位数, 32位、64位等;

工作频率: 工作时钟频率MHz为单位。

第10题: 计算机中机械硬盘的性能指标不包括( ); 其平均访问时间等于( )。

- A. 磁盘转速及容量
- B. 盘片数及磁道数
- C. 容量及平均寻道时间
- D. 磁盘转速及平均寻道时间

参考答案: B

解析:

本题考查计算机性能方面的基础知识。

硬盘的性能指标主要包括磁盘转速、容量、平均寻道时间。

第11题: 计算机中机械硬盘的性能指标不包括( ); 其平均访问时间等于( )。

- A. 磁盘转速 + 平均等待时间
- B. 磁盘转速 + 平均寻道时间
- C. 平均数据传输时间 + 磁盘转速
- D. 平均寻道时间 + 平均等待时间



参考答案：D

解析：

硬盘平均访问时间=平均寻道时间+平均等待时间。其中，平均寻道时间(Average seek time)是指硬盘在盘面上移动读写头至指定磁道寻找相应目标数据所用的时间，它描述硬盘读取数据的能力，单位为毫秒；平均等待时间也称平均潜伏时间 (Average latency time)，是指当磁头移动到数据所在磁道后，然后等待所要的数据块继续转动到磁头下的时间。

第12题：在互联网中，各种电子媒体按照超链接的方式组织，通常使用（ ）来描述超链接信息。

- A. HTML
- B. XML
- C. SGML
- D. VRML

参考答案：A

解析：

本题考查HTML基础知识。

HTML超文本标记语言，标准通用标记语言下的一个应用。

“超文本”就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。

XML是可扩展标记语言，标准通用标记语言的子集，是一种用于标记电子文件使其具有结构性的标记语言。

SGML标准通用标记语言（以下简称“通用标言”），是一种定义电子文档结构和描述其内容的国际标准语言；通用标言为语法置标提供了异常强大的工具，同时具有极好的扩展性，因此在数据分类和索引中非常有用；是所有电子文档标记语言的起源，早在万维网发明之前“通用标言”就已存在。

VRML ( Virtual Reality Modeling Language ) 即虚拟现实建模语言。是一种用于建立真实世界的场景模型或人们虚构的三维世界的场景建模语言，也具有平台无关性。

第13题：使用图像扫描仪以300DPI的分辨率扫描一幅3英寸×3英寸的图片，可以得到（ ）内存像素的数字图像。

- A. 100×100
- B. 300×300
- C. 600×600
- D. 900×900

参考答案：D

解析：

本题考查多媒体基础知识。

图像分辨率是指一幅图像的像素密度，每英寸多少点 (dpi) 表示图像大小  
如果200dpi扫描一幅2\*2.5英寸的照片，则可以得到400\*500像素点的图像。

第14题：根据《计算机软件保护条例》的规定，当软件 ( ) 后，其软件著作权才能得到保护。

- A. 作品发表
- B. 作品创作完成并固定在某种有形物体上
- C. 作品创作完成
- D. 作品上加注版权标记

参考答案：C

解析：

本题考查知识产权保护方面的基本知识。

根据《中华人民共和国著作权法》和《计算机软件保护条例》的规定，计算机软件著作权的权利自软件开发完成之日起产生，公民的软件著作权保护期为公民终生及其死亡之后50年；法人或其他组织的软件著作权保护期为50年。保护期满，除开发者身份权以外，其他权利终止。一旦计算机软件著作权超出保护期后，软件进入公有领域。计算机软件著作权人的单位终止和计算机软件著作权人的公民死亡均无合法继承人的，除开发者身份权以外，该软件的其他权利进入公有领域。软件进入公有领域后成为社会公共财富，公众可无偿使用。

第15题：甲程序员为乙软件设计师开发的应用软件编写了操作说明书，并已交付用户使用， ( ) 享有该应用软件的软件著作权。

- A. 甲程序员
- B. 甲程序员和乙软件设计师共同
- C. 乙软件设计师
- D. 甲程序员和乙软件设计师都不

参考答案：B

解析：

《著作权法实施条例》第四条规定了作品范围，虽然没有明确把“操作说明书”列入作品范围，但操作说明书中文学性的说明、产品设计图以及其他的一些图片、标识只要是符合著作权法关于作品的实质条件，即独创性，就可以作为作品予以保护。在这里操作说明书是对产品使用及性能独有的文字描述，以文字、照片、示意图的组合表达产品特定的内容。企业在生产经营中，为了介绍产品、宣传企业，引导用户额操作操作，经过智力劳动，设计创作了



操作说明书, 以照片、表格、文字叙述构成的整体设计融入了设计者的智慧, 并付出一定的金钱和劳动, 因此操作说明书的创作者也应该享受著作权。

第16题: 数字信封技术能够 ( )。

- A. 保证数据在传输过程中的安全性
- B. 隐藏发送者的真实身份
- C. 对发送者和接收者的身份进行认证
- D. 防止交易中的抵赖发生

参考答案: A

解析:

本题考查加密技术中数字信封技术的基本知识。

数字信封是将对称密钥通过非对称加密 (即: 有公钥和私钥两个) 的结果分发对称密钥的方法。数字信封是实现信息完整性验证的技术。

数字信封是一种综合利用了对称加密技术和非对称加密技术两者的优点进行信息安全传输的一种技术。数字信封既发挥了对称加密算法速度快、安全性好的优点, 又发挥了非对称加密算法密钥管理方便的优点。

数字信封以发送方向接收方传递一段交易信息 (如电子合同、支付通知单等) 为例, 发送方先在本地用对称密钥对交易信息进行加密, 形成密文, 再用接收方的公钥将用于加密交易信息的对称密钥加密, 并将加密后的对称密钥信息和密文一同传递给接收方。接收方接收信息后, 先用自己的私钥解密加密的对称密钥信息, 得到用于加密交易信息的对称密钥, 再用其解密密文得到交易信息原文。由于在传递过程中, 加密后的对称密钥就像是被封装在一个 "信封" 里传递一样, 因此被称为数字信封。

第17题: ( ) 属于DoS攻击。

- A. 为Wi-Fi设置安全连接口令, 拒绝用户访问网络
- B. 设置访问列表从而拒绝指定用户访问
- C. 发送大量非法请求使得服务器拒绝正常用户的访问
- D. 为用户设定相应的访问权限

参考答案: C

解析:

本题考查网络安全基础知识。

DoS是Denial of Service的简称, 即拒绝服务, 造成DoS的攻击行为被称为DoS攻击, 其目的是使计算机或网络无法提供正常的服务。最常见的DoS攻击有计算机网络带宽攻击和连通性攻击。

DoS攻击是指故意的攻击网络协议实现的缺陷或直接通过野蛮手段残忍地耗尽被攻击对象的

资源, 目的是让目标计算机或网络无法提供正常的服务或资源访问, 使目标系统服务系统停止响应甚至崩溃, 而在此攻击中并不包括侵入目标服务器或目标网络设备。

第18题: 在网络安全管理中, 加强内防内控可采取的策略有 ( )。

- 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

- A. ②③
- B. ②④
- C. ①②③④
- D. ②③④

参考答案: C

解析:

本题考查网络安全基础知识。

内网安全系统紧密围绕"合规", 内含企业级主机防火墙系统, 通过"终端准入控制、终端安全控制、桌面合规管理、终端泄密控制和终端审计"五维化管理, 全面提升内网安全防护能力和合规管理水平。因此1、2、3、4都属于内防内控的范围。

第19题: 某计算机内存空间按字节编址, 起始地址为0000A000H、终止地址为0000BFFFH的内存区域容量为 ( ) KB。

- A. 8
- B. 13
- C. 1024
- D. 8192

参考答案: A

解析:

本题考查计算机硬件中的内存部件基础知识。

存储区域空间为:  $0000BFFF - 0000A000 + 1 = 2000H$ , 按字节编址。

总容量为:  $2 \times 16^3 = 8 \times 2^{10} = 8K$ 。

第20题: 计算机在进行浮点数的相加 (减) 运算前需先进行对阶操作, 若x的阶码大于y的阶码, 则应将 ( )。

- A. x的阶码缩小至与y的阶码相同, 并对x的尾数进行算术左移

- B. x的阶码缩小至与y的阶码相同, 并对x的尾数进行算术右移
- C. y的阶码扩大至与x的阶码相同, 并对y的尾数进行算术左移
- D. y的阶码扩大至与x的阶码相同, 并对y的尾数进行算术右移

参考答案: D

解析:

在浮点数加减运算时, 首先要进行对阶, 根据对阶的规则, 阶码和尾数将进行相应的操作。

对阶, 首先应求出两数阶码 $E_x$ 和 $E_y$ 之差, 即 $\Delta E = E_x - E_y$

若 $\Delta E = 0$ , 表示两数阶码相等, 即 $E_x = E_y$ ; 若 $\Delta E > 0$ , 表示 $E_x > E_y$ ; 若 $\Delta E < 0$ , 表示 $E_x < E_y$ 。

当 $E_x \neq E_y$ 时, 要通过尾数的移位来改变 $E_x$ 或 $E_y$ , 使 $E_x = E_y$ 相等。对阶的规则是: 小阶向大阶看齐。即阶码小的数的尾数右移, 每右移一位, 阶码加1, 直到两数的阶码相等为止。如:

$E_x = E_y$ , 无需对阶。

$E_x > E_y$ , 则 $M_y$ 右移。每右移一位,  $E_y + 1 \rightarrow E_y$ , 直至 $E_x = E_y$ 为止。

$E_x < E_y$ , 则 $M_x$ 右移。每右移一位,  $E_x + 1 \rightarrow E_x$ , 直至 $E_x = E_y$ 为止。

第21题: 与 $X \oplus Y + X \cdot \bar{Y}$  等价的逻辑表达式为 ( )。

- A.  $X + Y$
- B.  $X$
- C.  $\bar{Y}$
- D.  $\bar{X} + \bar{Y}$

参考答案: B

解析:

本题考查数据的逻辑运算。

X	Y	$XY + X\bar{Y}$	$X + Y$	X	$\bar{Y}$	$\bar{X} + \bar{Y}$
1	0	1	1	1	1	1
1	1	1	1	1	0	0
0	0	0	0	0	1	1
0	1	0	1	0	0	1

第22题: 设机器码的长度为8位, 已知 $[X]_{\text{原}} = [Y]_{\text{补}} = 1111\ 1111$ , 且X和Y都是带符号的整数, 则X和Y的真值分别为 ( )。

- A. -127, -127

B. -1,-127

C. -127,-1

D. -1,-1

参考答案：C

解析：

本题考查数据的表示。

其中-127的原码是1111 1111。而-1的补码表示为：1111 1111。

	数值1	数值-1	1-1
原码	0000 0001	1000 0001	1000 0010
反码	0000 0001	1111 1110	1111 1111
补码	0000 0001	1111 1111	0000 0000
移码	1000 0001	0111 1111	1000 0000

第23题：以下关于Windows系统中文件的叙述中，正确的是（ ）。

- A. 文件一旦保存后则不能被删除
- B. 文件必须占用磁盘的连续区域
- C. “xls”也是可执行文件的扩展名
- D. 不同文件夹下的文件允许同名

参考答案：D

解析：

本题考查文件管理基础知识。

文件是允许删除和修改的；文件的可以按照索引等方式进行存储，存储在不连续的存储区域里；xls是Excel文件；不同文件夹下的文件允许同名，但是相同文件夹里面的文件名不允许相同，且不区分大小写。

第24题：在Windows资源管理器中，若要选择窗口中分散的多个文件，在缺省配置下，可以先选择一个文件，然后按住（ ）。

- A. CTRL键不放，并用鼠标右键单击要选择的文件
- B. CTRL键不放，并用鼠标左键单击要选择的文件
- C. SHIFT键不放，并用鼠标右键单击要选择的文件
- D. SHIFT键不放，并用鼠标左键单击要选择的文件

参考答案：B

解析：

本题考查文件管理基础知识。

如果要选择多个连续的文件，可以使用按住Shift进行点选第一个和最后一个即可，如果是分散的文件，则可以使用按住Ctrl键不放，然后左键单击选择需要选择的文件。

第25题：假设某计算机系统中资源R的可用数为9，系统中有3个进程竞争R，且每个进程都需要*i*个R，则该系统可能会发生死锁的最小*i*值是（ ）。

A. 2

B. 3

C. 4

D. 5

参考答案：C

解析：

本题考查操作系统的基础知识。

有3个进程，每个进程都需要分配4个资源，如果先给每个进程都分配了3个资源，那么这时候没有可用资源，接下来不管怎么分配都会发生死锁。而如果每个进程需要3个资源，这个时候是满足需求的，不会发生死锁。

第26题：某计算机系统页面大小为4K，进程P的页面变换表如下图所示。若进程P要访问逻辑地址25EFH（十六进制），那么经过地址变换后物理地址应为（ ）。

页号	物理块号
0	1
1	3
2	4
3	6

A. 15EFH

B. 35EFH

C. 45EFH

D. 65EFH

参考答案：C

解析：

本题考查操作系统存储管理方面的基础知识。

根据题意, 页面大小为4K, 逻辑地址为十六进制25EFH其页号为2, 页内地址为5EFH, 查页表后可知页帧号(物理块号)为4, 该地址经过变换后, 其物理地址应为页帧号4拼上页内地址5EFH, 即十六进制45EFH。

第27题：嵌入式操作系统主要特点包括：微型化、( )。

- A. 可定制、实时、可靠, 但不易移植
- B. 可定制、实时、可靠和易移植
- C. 实时、可靠、易移植, 但不可定制
- D. 可定制、实时、易移植性, 但可靠性差

参考答案：B

解析：

本题考查操作系统基础知识。

嵌入式操作系统运行在智能芯片环境中, 其特点有：微型化、可定制、实时性、可靠性、易移植性。

第28题：( ) 不属于脚本语言。

- A. COBOL
- B. Python
- C. PHP
- D. JavaScript

参考答案：A

解析：

本题考查程序设计语言基础知识。

主要的程序语言如下：

Fortran语言(第一个高级程序设计语言, 科学计算, 执行效率高)

Pascal语言(结构化程序设计语言, 表达能力强, Delphi)

C语言(通用、结构化程序设计语言, 指针操作能力强, 高效)

Lisp语言(函数式程序语言, 符号处理, 人工智能)

C++语言(C语言基础上增加了类机制, 面向对象, 高效)



Java语言 (面向对象, 中间代码, 跨平台, 通用的程序设计语言)  
Python (面向对象, 解释型程序设计语言, 通用的脚本语言)  
PHP (服务器端脚本语言, 制作动态网页)  
Ruby (简单快捷、面向对象、脚本语言)  
Delphi (快速应用程序开发工具, 可视化编程环境)  
COBOL (数据处理领域最为广泛的程序设计语言, 高级编程语言)  
PROLOG (逻辑式语言, 建造专家系统、自然语言理解、智能知识库等)

第29题: 将某高级语言程序翻译为汇编语言形式的目标程序, 该过程称为 ( )。

- A. 编译
- B. 解释
- C. 汇编
- D. 解析

参考答案: A

解析:

本题考查编译原理。

编译程序也叫编译系统, 是把用高级语言编写的面向过程的源程序翻译成目标程序的语言处理程序。编译程序把一个源程序翻译成目标程序的工作过程分为五个阶段: 词法分析; 语法分析; 中间代码生成; 代码优化; 目标代码生成。其主要是进行词法分析和语法分析, 又称为源程序分析, 分析过程中发现有语法错误, 给出提示信息, 然后再通过连接程序将编译后的目标文件连接成可执行的应用程序。

第30题: 在Windows系统中, 执行x.exe时系统报告找不到y.dll, 原因是 ( )。

- A. 程序x中存在语法或语义错误, 需要修改与x对应的源程序
- B. 程序y中存在语法错误, 需要修改与y对应的源程序
- C. 程序y中存在语义错误, 需要修改与y对应的源程序并重新编译
- D. 程序x执行时需要调用y中的函数。要安装y.dll

参考答案: D

解析:

本题考查windows基本知识。

DLL(Dynamic Link Library)文件为动态链接库文件, 又称“应用程序拓展”, 是软件文件类型。在Windows中, 许多应用程序并不是一个完整的可执行文件, 它们被分割成一些相对独立的动态链接库, 即DLL文件, 放置于系统中。当我们执行某一个程序时, 相应的DLL文件就会被调用。DLL文件中存放的是各类程序的函数(子过程)实现过程, 当程序需要调用函数时需要先载入DLL, 然后取得函数的地址, 最后进行调用。本题在执行x.exe文件时找不到y.dll,

说明该程序在运行时需要调用y.dll文件, 而找不到该文件, 就需要安装该文件以使程序运行。

第31题: 算术表达式采用后缀式表示时不需要使用括号, 使用 ( ) 就可以方便地进行求值。a - b \* ( c + d ) 的后缀式为 ( )。

- A. 队列
- B. 数组
- C. 栈
- D. 广义表

参考答案: C

解析:

本题考查编译原理基础知识。

计算机在处理算术表达式时, 首先将其转换为后缀表达式。例如, 表达式 "46+5\*(120-37)" 的后缀表达式形式为 "46 5 120 37 - \* + "。计算后缀表达式时, 从左至右扫描后缀表达式: 若遇到运算对象, 则压入栈中; 遇到运算符, 则从栈中弹出相关运算对象进行计算, 并将运算结果压入栈中, 重复以上过程, 直到后缀表达式扫描结束。

第32题: 算术表达式采用后缀式表示时不需要使用括号, 使用 ( ) 就可以方便地进行求值。a - b \* ( c + d ) 的后缀式为 ( )。

- A. a b c d - \* +
- B. a b c d \* + -
- C. a b - c \* d +
- D. a b c d + \* -

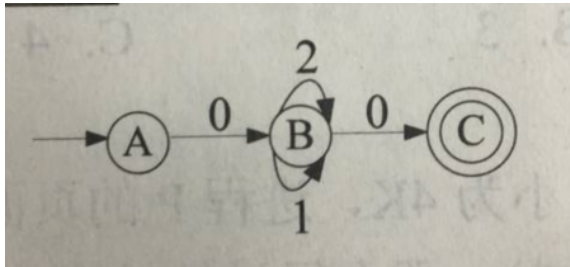
参考答案: D

解析:

表达式 "a-b\*(b+d)" 的后缀表达式形式为 "abcd + \* -"。

一种比较简便的方法: a - b \* ( c + d ), 看计算的先后顺序, 先是c+d, 转化的时候将操作数放前面, 然后再是运算符放后面, 注意操作数的前后顺序保持不变, 所以是cd+。然后b \* ( c + d ), 同样, 操作数放前面, 运算符放后面, 所以是bcd+\*。最后是a - b \* ( c + d ), 操作数放前面, 运算符放后面, 最后就是abcd+\*-。

第33题: 下图是一个有限自动机的状态转换图 ( A为初态、C为终态 ), 该自动机识别的字符串集合可用正规式 ( ) 来表示。



- A.  $(1|2)^*00$
- B.  $0(1|2)^*0$
- C.  $(0|1|2)^*$
- D.  $00(1|2)^*$

参考答案：B

解析：

本题考查程序语言基础知识。

一个有限自动机所识别的语言是从开始状态到终止状态所有路径上的字符串的集合。要判断一个字符串能否被指定的自动机识别，就看在该自动机的状态图中能否找到一条从开始状态到达终止状态的路径，且路径上的字符串等于需要识别的字符串。

从图中看，首先要识别字符0，然后最终要识别的也是字符0，中间识别1或者2，可以0次或者无穷次。

第34题：已知函数f1()、f2()的定义如下，调用f1时为传值调用，调用f2时为引用调用。设有函数调用t = f1(5)，那么调用执行后t的值为( )。

int f1(int a)	int f2(int &x)
<pre>int x = 1; a = f2(x); return a+x;</pre>	<pre>if (x&gt;1) { x = x+1; return 0; } else { x = x - 1; return 1; }</pre>

- A. 1
- B. 5
- C. 6
- D. 7

参考答案：A

解析：

本题考查程序语言基本知识。

首先 $a=f2(x)=f2(1)$ , 代入到 $f2$ 中进行运算,  $x$ 的值为0, 返回值为1。这里 $x$ 是引用调用, 所以 $x$ 的终值为0, 然后返回值赋给了 $a$ 。所以 $a=1$ 。最终 $t$ 的值为 $a+x=1$ 。

第35题: 设数组 $a[1..m, 1..n]$  ( $m > 1, n > 1$ ) 中的元素按行存放, 每个元素占用1个存储单元, 则数组元素 $a[i, j]$  ( $1 \leq i \leq m, 1 \leq j \leq n$ ) 相对于数组首元素的偏移量为 ( )。

- A.  $(i-1)*m+j-1$
- B.  $(i-1)*n+j-1$
- C.  $(j-1)*m+i-1$
- D.  $(j-1)*n+i-1$

参考答案: B

解析:

本题考查数组的存储。

存储数组元素时, 需要将元素按照某种顺序排列。对于二维及多维数组, 则有按行存储和按列存储两种方式, 其不同在于同一个元素相对于数组空间起始位置的偏移量不同。本问题中 $n$ 行 $m$ 列的二维数组 $a[1..m, 1..n]$ 是按行存储, 则对于元素 $a[i, j]$ 来说, 它之前有完整的 $i-1$ 列、每行 $n$ 个元素, 在第 $i$ 行上排在 $a[i, j]$ 之前的元素个数是 $j-1$ 个, 因此排列在 $a[i, j]$ 之前的元素个数为 $(i-1)*n+j-1$ , 由于每个元素占一个单元, 该表达式的值就是偏移量。

第36题: 一个计算机算法是对特定问题求解步骤的一种描述。算法的 ( ) 是指算法能够对不合理数据及非法操作进行识别和处理的能力。

- A. 有穷性
- B. 可行性
- C. 确定性
- D. 健壮性

参考答案: D

解析:

本题考查算法的基本知识。

评定一个算法的优劣, 主要有以下几个指标。

(1) 正确性: 一个算法必须正确才有存在的意义, 这是最重要的指标, 要求编程人员应用正确的计算机语言实现算法的功能。

(2) 友好性: 算法实现的功能是给用户使用的, 自然要具有良好的使用性, 即用户友好性。

(3) 可读性: 算法的实现可能需要多次的修改, 也可能被移植到其他的功能中, 因此算法应当是可读的、可以理解的, 方便程序人员对其分析、修改移植到自己的程序中, 实现某些功能。

(4) 健壮性: 在一个算法中, 经常会出现不合理的数据或非法的操作, 所以一个算法必须具

有健壮性, 能够对这些问题进行检查、纠正。算法具有健壮性是一个升华, 当用户刚开始学习写算法时可以忽略它的存在, 在逐渐的学习中要努力让算法更加完美。

(5) 效率: 算法的效率主要是指执行算法时计算机资源的消耗, 包括计算机内存的消耗和计算机运行时间的消耗。这两个消耗可以统称为时空效率。一个算法只有正确性而无效率是没有意义的, 通常, 效率也可以评定一个算法是否正确。如果一个算法需要执行几年甚至几百年, 那么无疑这个算法会被评为是错误的。

第37题: 假设以S和X分别表示入栈和出栈操作, 并且初始和终止时栈都为空, 那么( )不是合法的操作序列。

- A. SSXXXSSXSX
- B. SSSXXXSSXX
- C. SSXSSXSXXX
- D. SXSXSXSXSX

参考答案: A

解析:

本题考察栈和队列基本知识。

栈是先进后出, 在A选项中, SSXXX这里, 先进入了两个元素, 然后出栈3个, 此时就会产生错误, 因此是不合法的操作序列。

第38题: 设有一份电文中共使用a、b、c、d、e、f这6个字符, 它们的出现频率如下表所示, 现通过构造哈夫曼树为这些字符编码。那么, 编码长度最长的两个字符是( )。

字符	a	b	c	d	e	f
频率	0.19	0.05	0.23	0.13	0.34	0.06

- A. c、e
- B. b、e
- C. b、f
- D. e、f

参考答案: C

解析:

本题考查哈夫曼树构造方法。

构造最优二叉树的哈夫曼算法如下。



① 根据给定的 $n$ 个权值  $\{W_1, W_2, \dots, W_n\}$  构成 $n$ 棵二叉树的集合 $F = \{T_1, T_2, \dots, T_n\}$ , 其中每棵树 $T$ 中只有一个带权为 $W$ 的根结点, 其左右子树均空。

② 在 $F$ 中选取两棵根结点的权值最小的树作为左右子树, 构造一棵新的二叉树, 置新构造二叉树的根结点的权值为其左、右子树根结点的权值之和。

③ 从 $F$ 中删除这两棵树, 同时将新得到的二叉树加入到 $F$ 中。

重复②、③, 直到 $F$ 中只含一棵树时为止。这棵树便是最优二叉树 (哈夫曼树)。从树中一个结点到另一个结点之间的分支构成这两个结点之间的路径, 路径上的分支数目称为路径长度。树的路径长度是从树根到每一个结点的路径长度之和。树的带权路径长度为树中所有叶子结点的带权路径长度之和。

根据算法, 那么最长的路径应该就是b、f。

第39题: 对二叉树进行后序遍历和中序遍历时, 都依照左子树在前右子树在后的顺序。已知对某二叉树进行后序遍历时, 结点M是最后被访问的结点, 而对其进行中序遍历时, M是第一个被访问的结点, 那么该二叉树的树根结点为M, 且 ( )。

- A. 其左子树和右子树都必定为空
- B. 其左子树和右子树都不为空
- C. 其左子树必定为空
- D. 其右子树必定为空

参考答案: C

解析:

本题考查二叉树的基本知识。

前序遍历: 先遍历根结点, 然后遍历左子树, 最后遍历右子树。

中序遍历: 先遍历左子树, 然后遍历根结点, 最后遍历右子树。

后序遍历: 先遍历左子树, 然后遍历右子树, 最后遍历根节点。

结点M是树根结点, 而在中序遍历的时候, M是第一个被访问的结点, 那么可以看出其左子树应该为空。

第40题: 某图的邻接矩阵如下, 该图为 ( ) ; 若采用邻接表表示该图, 则邻接表中用来表示边 (或弧) 的表结点总数为 ( ) 个。

$\infty$	23	$\infty$	$\infty$	18	$\infty$
$\infty$	$\infty$	7	$\infty$	$\infty$	$\infty$
14	$\infty$	$\infty$	8	16	$\infty$
$\infty$	5	$\infty$	$\infty$	$\infty$	20
$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	12
$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$

- A. 无向图



- B. 有向图
- C. 完全图
- D. 二部图

参考答案：B

解析：

本题考查数据结构基础知识。

图的邻接矩阵是一个方阵，所有行标和列标都与图中的顶点一一对应，这样对于矩阵中的一个元素  $[i, j]$ ，其值为1 表示  $i, j$  对应的顶点间有边(或弧)，其值为 0则表示  $i, j$ 对应的顶点间不存在边(或弧)。显然，图中总共有9条边。

在无向图中，边  $(i, j)$ 与 $(j, i)$ 是指同一条边，其取值是相同的；在有向图中， $\langle i, j \rangle$ 与 $\langle j, i \rangle$ 是两条不同的弧。而在本题中，矩阵中的 $(i, j)$ 与 $(j, i)$ 是不同的，因此这个是有向图。

第41题：某图的邻接矩阵如下，该图为（ ）；若采用邻接表表示该图，则邻接表中用来表示边（或弧）的表结点总数为（ ）个。

$\infty$	23	$\infty$	$\infty$	18	$\infty$
$\infty$	$\infty$	7	$\infty$	$\infty$	$\infty$
14	$\infty$	$\infty$	8	16	$\infty$
$\infty$	5	$\infty$	$\infty$	$\infty$	20
$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	12
$\infty$	$\infty$	$\infty$	$\infty$	$\infty$	$\infty$

- A. 9
- B. 18
- C. 21
- D. 49

参考答案：A

解析：

本题考查数据结构基础知识。

图的邻接矩阵是一个方阵，所有行标和列标都与图中的顶点一一对应，这样对于矩阵中的一个元素  $[i, j]$ ，其值为1 表示  $i, j$  对应的顶点间有边(或弧)，其值为 0则表示  $i, j$ 对应的顶点间不存在边(或弧)。显然，图中总共有9条边。

在无向图中，边  $(i, j)$ 与 $(j, i)$ 是指同一条边，其取值是相同的；在有向图中， $\langle i, j \rangle$ 与 $\langle j, i \rangle$ 是两条不同的弧。而在本题中，矩阵中的 $(i, j)$ 与 $(j, i)$ 是不同的，因此这个是有向图。

第42题：若关键码序列 ( 47, 61, 55, 39, 10, 26, 90, 82 ) 采用散列法进行存储和查找。设散列函数为  $H(\text{Key}) = \text{Key} \bmod 11$  ( mod表示整除取余运算 ), 拟采用链地址法 ( 拉链法 ) 解决冲突构造散列表。以下关于该散列表的叙述中, 正确的是 ( )。

- A. 关键码10和90位于同一个链中
- B. 关键码61和82位于同一个链中
- C. 关键码61和39位于同一个链中
- D. 关键码47、55和39位于同一个链中

参考答案：C

解析：

本题考查数据结构和算法基本知识。

散列函数为  $H(\text{Key}) = \text{Key} \bmod 11$  (MOD表示整除取余运算), 因此只需要对线性表类数据分别与11进行取余运算。

分别将关键码序列和11进行取余运算, 得到 { 3, 6, 0, 6, 10, 4, 2, 5 }, 可以看出关键码61和39的值是相同的, 因此其位于同一个链中。

第43题：用某排序方法对一个关键码序列进行递增排序时, 对于其中关键码相同的元素, 若该方法可保证在排序前后这些元素的相对位置不变, 则称该排序方法是稳定的。以下关于排序方法稳定性的叙述中, 正确的是 ( )。

- A. 冒泡排序和简单选择排序都是稳定的排序方法
- B. 冒泡排序是稳定的排序方法, 简单选择排序不是
- C. 简单选择排序是稳定的排序方法, 冒泡排序不是
- D. 冒泡排序和简单选择排序都不是稳定的排序方法

参考答案：B

解析：

本题考查排序算法。

主要的排序时间复杂度及其稳定性如下：

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	$O(n^{1.25})$	-----	$O(1)$	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

第44题：多态分为参数多态、包含多态、过载多态和强制多态四种不同形式。其中，子类型化（一个类型是另一个类型的子类型）属于（ ）多态；（ ）多态是指同一个名字在不同上下文中可代表不同的含义。

- A. 参数
- B. 包含
- C. 过载
- D. 强制

参考答案：B

解析：

本题考查面向对象技术知识。

多态指的是不同对象对同一消息的不同的响应。主要分为：

参数多态：应用广泛、最纯的多态。

包含多态：同样的操作可用于一个类型及其子类型。包含多态一般需要进行运行时的类型检查。包含多态在许多语言中都存在,最常见的例子就是子类型化,即一个类型是另外一个类型的子类型。

强制多态：编译程序通过语义操作，把操作对象的类型强行加以变换，以符合函数或操作符的要求。

过载多态：同一个名（操作符、函数名）在不同的上下文中有不同的类型。

第45题：多态分为参数多态、包含多态、过载多态和强制多态四种不同形式。其中，子类型化（一个类型是另一个类型的子类型）属于（ ）多态；（ ）多态是指同一个名字在不同上下文中可代表不同的含义。

- A. 参数
- B. 包含

C. 过载

D. 强制

参考答案：C

解析：

本题考查面向对象技术知识。

多态指的是不同对象对同一消息的不同的响应。主要分为：

参数多态：应用广泛、最纯的多态。

包含多态：同样的操作可用于一个类型及其子类型。包含多态一般需要进行运行时的类型检查。包含多态在许多语言中都存在,最常见的例子就是子类型化,即一个类型是另外一个类型的子类型。

强制多态：编译程序通过语义操作,把操作对象的类型强行加以变换,以符合函数或操作符的要求。

过载多态：同一个名(操作符、函数名)在不同的上下文中有不同的类型。

第46题：UML中的结构事物是模型中的静态部分,采用名词描述概念或物理元素。( )属于结构事物,以此事物为主的UML模型图为( )。

A. 组件

B. 状态机

C. 注释

D. 包

参考答案：A

解析：

本题考查UML基础知识。

事物：是对模型中最具有代表性的成分的抽象。

结构事物,如类(Class)、接口(Interface)、协作(Collaboration)、用例(UseCase)、主动类(ActiveClass)、组件(Component)和节点(Node);

行为事物,如交互(Interaction)、状态机(State machine);

分组事物(包,Package);

注释事物(注解,Note)。

第47题：UML中的结构事物是模型中的静态部分,采用名词描述概念或物理元素。( )属于结构事物,以此事物为主的UML模型图为( )。

A. 类图

B. 组件图

C. 包图

## D. 部署图

参考答案：B

解析：

本题考查UML基础知识。

事物：是对模型中最具有代表性的成分的抽象。

结构事物，如类 ( Class )、接口 ( Interface )、协作 ( Collaboration )、用例 ( UseCase )、主动类 ( ActiveClass )、组件 ( Component ) 和节点 ( Node )；

行为事物，如交互 ( Interaction )、状态机 ( Statemachine )；

分组事物 ( 包, Package )；

注释事物 ( 注解, Note )。

第48题：创建型设计模式抽象了实例化过程，它分为面向类和面向对象两种。其中，( ) 模式是保证一个类仅有一个实例，并提供一个访问它的全局访问点。

- A. 工厂 ( Factory )
- B. 构建器 ( Builder )
- C. 原型 ( Prototype )
- D. 单例 ( Singleton )

参考答案：D

解析：

本题考查设计模式。

单例模式有以下特点：1、单例类只能有一个实例。2、单例类必须自己创建自己的唯一实例。3、单例类必须给所有其他对象提供这一实例。

单例模式确保某个类只有一个实例，而且自行实例化并向整个系统提供这个实例。在计算机系统中，线程池、缓存、日志对象、对话框、打印机、显卡的驱动程序对象常被设计成单例。这些应用都或多或少具有资源管理器的功能。每台计算机可以有若干个打印机，但只能有一个Printer Spooler，以避免两个打印作业同时输出到打印机中。每台计算机可以有若干通信端口，系统应当集中管理这些通信端口，以避免一个通信端口同时被两个请求同时调用。总之，选择单例模式就是为了避免不一致状态，避免政出多头。

第49题：以下关于软件工程的叙述中，错误的是( )。

- A. 软件工程是开发、运行、维护软件的系统方法
- B. 软件工程把系统的、规范的、可度量的途径应用于软件开发、运行、维护过程
- C. 软件工程将工程技术、工程管理和工程经济融入软件开发过程
- D. 软件工程就是以工业流水线生产方式开发软件的过程

参考答案：D

解析：

本题考查软件工程基础知识。

现阶段的软件开发方法有很多种，并不都是以流水线生产方式。

第50题：用户界面设计的原则不包括（ ）。

- A. 将用户界面置于用户控制之下
- B. 减少用户的记忆负担
- C. 用户界面上应使用统一的IT术语
- D. 保持界面简洁和平衡

参考答案：C

解析：

本题考查UI用户界面的设计。

用户界面设计原则：

简易性（便于使用和理解，减少错误选择的可能）

用户的语言（友好人性化提示，使用用户容易接受的方式）

记忆负担最小化（转换界面而不是选择一系列的命令行）

一致性（术语一致、风格和内容一致、色调字体一致等）

利用用户的熟悉程度（与其他众多的字体风格、按钮颜色等一致）

从用户的观点考虑（以用户为中心，用户参与界面设计）

排列分组（有序整齐方便查找和使用）

安全性（选择可逆，避免选错无法恢复）

人性化（从熟练用户和新手用户双方考虑，尽量可以根据需求定制界面）

第51题：敏捷开发的原则不包括（ ）。

- A. 使用分阶段的计划来管理流程，并详细地编写开发文档
- B. 欢迎需求变化，并利用这种变化来提高用户的竞争优势
- C. 尽早并持续地发布并交付有价值的软件以满足用户需求
- D. 无论团队内外，面对面的交流始终是最有效的沟通方式

参考答案：D

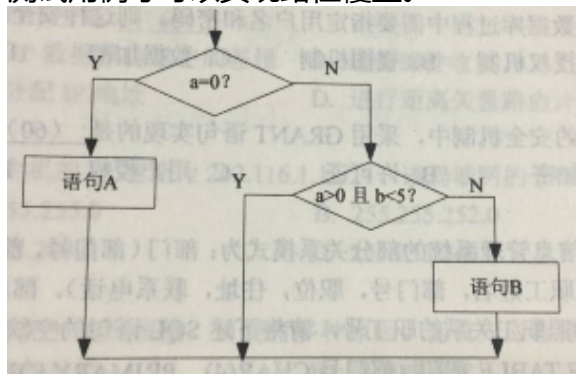
解析：

本题考查敏捷开发方法。



关于团队内外的交流沟通, 如果都采用面对面交流就会花费大量的时间和精力, 因此要采用合适的方式进行沟通交流。

第52题: 在设计白盒测试用例时, ( ) 是最弱的逻辑覆盖。在下图所示流程中, 至少需要 ( ) 个测试用例才可以实现路径覆盖。



- A. 条件覆盖
- B. 语句覆盖
- C. 判定覆盖
- D. 路径覆盖

参考答案: B

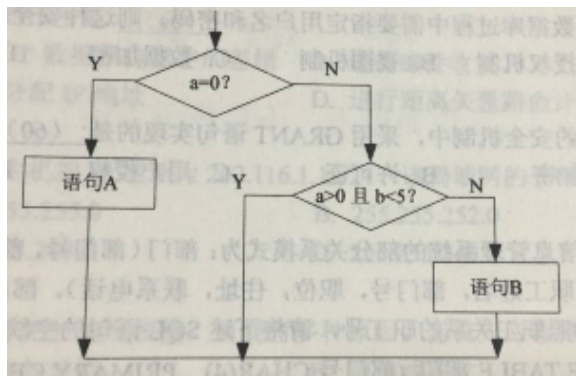
解析:

本题考查软件测试的基本知识。

主要的软件测试方法有:

- (1) 语句覆盖。被测程序的每个语句至少执行一次。是一种很弱的覆盖标准。
  - (2) 判定覆盖。也称为分支覆盖, 判定表达式至少获得一次“真”、“假”值。判定覆盖比语句覆盖强。
  - (3) 条件覆盖。每个逻辑条件的各种可能的值都满足一次。
  - (4) 路径覆盖。覆盖所有可能的路径。
  - (5) 判定/条件覆盖。每个条件所有可能的值(真/假)至少出现一次。
  - (6) 条件组合覆盖。每个条件的各种可能值的组合都至少出现一次。
- 满足路径覆盖, 只要覆盖所有可能的路径, 总共是3条路径, 所以需要3个测试用例即可。

第53题: 在设计白盒测试用例时, ( ) 是最弱的逻辑覆盖。在下图所示流程中, 至少需要 ( ) 个测试用例才可以实现路径覆盖。



- A. 1
- B. 2
- C. 3
- D. 4

参考答案：C

解析：

本题考查软件测试的基本知识。

主要的软件测试方法有：

- (1) 语句覆盖。被测程序的每个语句至少执行一次。是一种很弱的覆盖标准。
  - (2) 判定覆盖。也称为分支覆盖, 判定表达式至少获得一次“真”、“假”值。判定覆盖比语句覆盖强。
  - (3) 条件覆盖。每个逻辑条件的各种可能的值都满足一次。
  - (4) 路径覆盖。覆盖所有可能的路径。
  - (5) 判定/条件覆盖。每个条件所有可能的值(真/假)至少出现一次。
  - (6) 条件组合覆盖。每个条件的各种可能值的组合都至少出现一次。
- 满足路径覆盖, 只要覆盖所有可能的路径, 总共是3条路径, 所以需要3个测试用例即可。

第54题：以下关于软件可靠性测试的叙述中, 错误的是( )。

- A. 软件可靠性测试的目的是评估软件系统的可靠性
- B. 软件可靠性测试前应先确定软件可靠性的目标
- C. 应平衡地考虑对软件开发进度和成本的影响
- D. 应选用适用于所有软件的标准的可靠性测试模型

参考答案：D

解析：

本题考查软件测试基本知识。

没有适用于所有软件的标准的可靠性测试模型, 主要还是看软件本身的要求。

第55题: ( ) 模式将企业主要的数据处理过程从个人计算机或服务器转移到大型的数据中心, 将计算能力、存储能力当作服务来提供。

- A. 人工智能
- B. 物联网
- C. 云计算
- D. 移动互联网

参考答案: C

解析:

本题考查云计算基本知识。

云计算是利用高速互联网的传输能力, 将数据的处理过程从个人计算机或服务器转移到一个大型的计算中心, 并将计算能力、存储能力当作服务来提供。用户不再需要了解“云”中基础设施的细节, 不必具有相应的专业知识, 也无需直接进行控制, 就如同电力、自来水一样按需使用和按量计费。这便是云计算——“让地球更平”的计算方式。

第56题: 以下关于企业信息化的叙述中, 错误的是( )。

- A. 应从技术驱动的角度来构建企业一体化的信息系统
- B. 诸多信息孤岛催生了系统之间互联互通整合的需求
- C. 业务经常变化引发了信息系统灵活适应变化的需求
- D. 信息资源共享和业务协同将使企业获得更多的回报

参考答案: A

解析:

本题考查企业信息化建设基础知识。

一体化管理系统是指拥有多个企业管理模块的信息管理系统, 每个应用模块包含不同管理方向的功能, 如客户管理、采购管理、项目管理、OA、人力资源管理等, 通过一体化的设计架构, 实现企业数据共享。对于企业一体化的信息系统, 主要看的是企业的业务, 经营范围等等, 而不是考虑技术方面

第57题: 数据库的基本表与存储文件之间通过建立( )之间的映像, 保证数据的物理独立性。

- A. 外模式/模式
- B. 外模式/内模式
- C. 模式/内模式
- D. 外模式/外模式

参考答案：C

解析：

本题考查数据库的基本知识。

外模式/模式映象：定义在外模式描述中，把描述局部逻辑结构的外模式与描述全局逻辑结构的模式联系起来，保证逻辑独立性：当模式改变时，只要对外模式/模式映象做相应的改变，使外模式保持不变，则以外模式为依据的应用程序不受影响，从而保证了数据与程序之间的逻辑独立性，也就是数据的逻辑独立性。

模式/内模式映象：定义在模式描述中，把描述全局逻辑结构的模式与描述物理结构的内模式联系起来，保证物理独立性：当内模式改变时，比如存储设备或存储方式有所改变，只要模式/内模式映象做相应的改变，使模式保持不变，则应用程序保持不变。

第58题：事务的（ ）是指事务一旦提交，即使之后又发生故障，对其执行的结果也不会有任何影响。

- A. 原子性
- B. 持久性
- C. 隔离性
- D. 一致性

参考答案：B

解析：

本题考查事务的性质。

原子性：事务是原子的，要么做，要么都不做。

一致性：事务执行的结果必须保证数据库从一个一致性状态变到另一个一致性状态。

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

持久性：一旦事务成功提交，即使数据库崩溃，其对数据库的更新操作也永久有效。

串行调度：多个事务依次串行执行，且只有当一个事务的所有操作都执行完后才执行另一个事务的所有操作。

第59题：若连接数据库过程中需要指定用户名和密码，则这种安全措施属于（ ）。

- A. 授权机制
- B. 视图机制
- C. 数据加密
- D. 用户标识与鉴别

参考答案：D

解析：

本题考查信息安全的基本知识。

授权机制是指指定用户对数据库对象的操作权限；视图机制是通过视图访问而将基本表中视图外的数据对用户屏蔽实现安全；数据加密通过对存储和传输数据库的数据进行加密；用户标识与鉴别是指用户进入数据库系统时提供自己的身份标识，由系统鉴定是否为合法用户，只有合法用户才可以进入。

第60题：数据库的安全机制中，采用 GRANT语句实现的是（ ）。

- A. 加密
- B. 许可证
- C. 用户授权
- D. 回收权限

参考答案：C

解析：

本题考查数据库的安全机制。

在数据库中，GRANT是授予对数据库的权限。

第61题：某企业信息管理系统的部分关系模式为：部门（部门号，部门名，负责人，电话）、职工（职工号，职工姓名，部门号，职位，住址，联系电话）。部门关系中的部门名为非空值，负责人参照职工关系的职工号，请将下述SQL语句的空缺部分补充完整。

```
CREATE TABLE 部门 ( 部门号 CHAR(4) PRIMARY KEY ,  
                    部门名 CHAR(20) ( ),  
                    负责人 CHAR(6) ,  
                    电话 CHAR(15) ,  
                    ( ) ) ;
```

- A. UNIQUE
- B. NOT NULL
- C. KEY UNIQUE
- D. PRIMARY KEY

参考答案：B

解析：

本题考查SQL语言。

部门名前面有说明要求为非空, 因此用NOT NULL标识。

然后负责人参照职工关系的职工号, 职工号是外键。用FOREIGN KEY (负责人) REFERENCE 职工 (职工号) 表示。

第62题: 某企业信息管理系统的部分关系模式为: 部门 (部门号, 部门名, 负责人, 电话)、职工 (职工号, 职工姓名, 部门号, 职位, 住址, 联系电话)。部门关系中的部门名为非空值, 负责人参照职工关系的职工号, 请将下述SQL语句的空缺部分补充完整。

```
CREATE TABLE 部门 ( 部门号 CHAR(4) PRIMARY KEY ,  
                    部门名 CHAR(20) ( ),  
                    负责人 CHAR(6) ,  
                    电话 CHAR(15) ,  
                    ( ) ) ;
```

- A. PRIMARY KEY ( 部门号 ) NOT NULL UNIQUE
- B. PRIMARY KEY ( 部门名 ) UNIQUE
- C. FOREIGN KEY ( 负责人 ) REFERENCES 职工 ( 职工号 )
- D. FOREIGN KEY ( 负责人 ) REFERENCES 职工 ( 职工姓名 )

参考答案: C

解析:

本题考查SQL语言。

负责人参照职工关系的职工号, 职工号是外键。用FOREIGN KEY (负责人) REFERENCE 职工 (职工号) 表示。

第63题: 对于正实数a, 设[a]表示a的整数部分 (例如[3.14] = 3), 则 $[1/3] + [2/3] + [3/3] + \dots + [99/3] + [100/3] = ( )$ 。

- A. 1650
- B. 1682
- C. 1683
- D. 1749

参考答案: A

解析:

本题是基本计算问题。



$$[1/3]+[2/3]+[3/3]+.....+[99/3]+[100/3]=0+0+0+.....+32+32+32+33+33=(0+33)*17*3-33=1650。$$

第64题：某道多项选择题有A、B、C、D、E五个备选项，要求从中选择2项或2项以上，全部选对才算解答正确。按照上述要求，某考生凭猜测答题，解答正确的概率是（ ）。

- A. 1/10
- B. 1/16
- C. 1/26
- D. 1/31

参考答案：C

解析：

本题是基本计算问题。

答案的情况总共有

$$C_5^2 + C_5^3 + C_5^4 + C_5^5 = 10 + 10 + 5 + 1 = 26, \text{ 则解答正确的概率是 } 1/26。$$

第65题：甲乙两人同时从同一地点出发向相反方向沿同一条环形公路匀速行走，甲将用3小时走完一圈，乙将用2小时走完一圈，则他们将在出发后（ ）小时第一次相遇。

- A. 1.1
- B. 1.2
- C. 1.3
- D. 1.4

参考答案：B

解析：

本题是基本计算问题。

根据题意，甲的速度是1/3（圈/小时），乙的速度是1/2（圈/小时）。如果甲乙两人相遇，则两个人刚好加起来是走完一圈，那么就有：1/(1/3+1/2)=1.2小时。

第66题：在TCP/IP体系结构中，（ ）协议可将IP地址转化为MAC地址；（ ）协议属于应用层协议。

- A. RARP
- B. ARP

C. ICMP

D. TCP

参考答案：B

解析：

本题考查网络基础知识。

ARP是将IP地址转MAC地址，RARP是将MAC地址转IP地址。其中DNS是属于应用层的协议。

第67题：在TCP/IP体系结构中，（ ）协议可将IP地址转化为MAC地址；（ ）协议属于应用层协议。

A. UDP

B. IP

C. ARP

D. DNS

参考答案：D

解析：

本题考查网络基础知识。

ARP是将IP地址转MAC地址，RARP是将MAC地址转IP地址。其中DNS是属于应用层的协议。

第68题：ICMP协议的作用是（ ）。

A. 报告IP数据报传送中的差错

B. 进行邮件收发

C. 自动分配IP地址

D. 进行距离矢量路由计算

参考答案：A

解析：

本题考查网络基础知识。

ICMP是（ Internet Control Message Protocol ）Internet控制报文协议。它是TCP/IP协议族的一个子协议，用于在IP主机、路由器之间传递控制消息。控制消息是指网络通不通、主

机是否可达、路由是否可用等网络本身的消息。这些控制消息虽然并不传输用户数据,但是对于用户数据的传递起着重要的作用。

第69题:局域网中某主机的IP地址为202.116.1.12/21,该局域网的子网掩码为( )。

- A. 255.255.255.0
- B. 255.255.252.0
- C. 255.255.248.0
- D. 255.255.240.0

参考答案:C

解析:

本题考查子网掩码的基本知识。

202.116.1.12/21表示有21位为网络位,另外11位是主机位。因此,在子网掩码中,第三个字节的后面3位是0,因此子网掩码是255.255.248.0。

第70题:如果访问一个网站速度很慢,可能有多种原因,但首先应该排除的是( )。

- A. 网络服务器忙
- B. 通信线路忙
- C. 本地终端感染病毒
- D. 没有访问权限

参考答案:D

解析:

本题考查网络基础知识。

在这里,已经进入了一个网站,因此可以排除没有访问权限,如果没有访问权限的话,将不允许进入。

第71题:The term "software ( )" is used to describe the software engineering activities that occur following delivery of a software product to the customer.

- A. analysis
- B. maintenance
- C. design
- D. implementation

参考答案：B

解析：

本题考查计算机类英语知识。

术语“软件维护”是用来描述在向客户交付软件产品后发生的软件工程活动。

A分析 B维护 C设计 D开发

第72题：( ) testing ensures that a software module operates correctly and reliably.

A. Application

B. Integration

C. System

D. Unit

参考答案：D

解析：

本题考查计算机类英语知识。

单元测试确保软件模块正确和可靠地运行。

A应用 B集成 C系统 D单元

第73题：( ) refers to the range of computer hardware and OS platforms on which the source code of a program can be compiled and run.

A. Usability

B. Portability

C. Reliability

D. Maintainability

参考答案：B

解析：

本题考查计算机类英语知识。

可移植性指的是计算机硬件和操作系统平台的范围，在这些平台上都可以编译和运行程序的源代码。

A可使用性 B可移植性 C可靠性 D可维护性

第74题：A computer ( ) is a program that maliciously causes unwanted behavior on a computer.

- A. software
- B. application
- C. virus
- D. process

参考答案 : C

解析 :

本题考查计算机类英语知识。  
一个计算机病毒是一段代码, 它会对计算机造成不期望的恶性不良影响。  
A软件 B应用 C病毒 D过程

第75题 : ( ) makes computers able to do jobs which till now, only human brains can do.

- A. Artificial intelligence
- B. Big data
- C. Cloud computing
- D. Internet of things

参考答案 : A

解析 :

本题考查计算机类英语知识。  
人工智能能够让计算机做那些现在只有人类大脑才可以做的一些工作。  
A人工智能 B大数据 C云计算 D物联网

## 下午案例分析

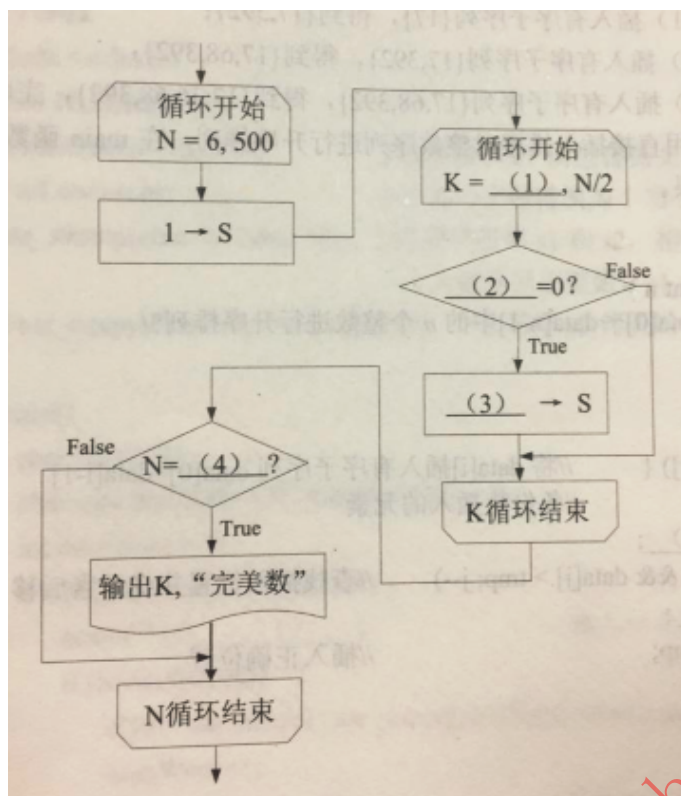
第1题 : 阅读以下说明和流程图, 填写流程图中的空缺, 将解答填入答题纸的对应栏内。

【说明】

如果一个自然数N恰好等于它所有不同的真因子 (即N的约数以及1, 但不包括N) 之和S, 则称该数为“完美数”。例如 $6 = 1 + 2 + 3$ ,  $28 = 1 + 2 + 4 + 7 + 14$ , 所以6和28都是完美数。显然, 6是第1个 (即最小的) 完美数。

下面流程图的功能是求500以内所有的完美数。

【流程图】



循环开始框中要注明：循环变量 = 初始值，终值，步长，步长为1时可以缺省。

如果某自然数小于其所有真因子之和（例如 $24 < 1 + 2 + 3 + 4 + 6 + 8 + 12$ ），则称该自然数为亏数；如果某自然数大于其所有真因子之和（例如 $8 > 1 + 2 + 4$ ），则称该自然数为赢数；如果某自然数等于从1开始的若干个连续自然数之和（例如 $10 = 1 + 2 + 3 + 4$ ）则称该自然数为三角形数。据此定义，自然数496是（ ）。

供选择答案：

- A. 亏数    B. 赢数    C. 完美数，非三角形数    D. 完美数和三角形数

题：1.1

答案解析：

【参考答案】

- (1) 2  
(2)  $N \% K$   
(3)  $S + K$   
(4) S  
(5) D

【解析】

流程图的功能是求500以内所有的完美数，N的值范围是6~500，因此N是需要判断是否为完美数，首先要求出N的所有真因子，然后再判断N和真因子之和是否相等，从流程图可以看出S是保存真因子和的变量，K是保存真因子的变量，因此K的初始值是2，终值是N/2，因此第（1）空处填写：2；判断K是否为N的真因子，即判断 $N \% K$ （N除以K取余）是否为0，



第(2)空填写:  $N \% K$ ; 当K为N的真因子时, 需要计算所有K的和, 即  $S = S + K$ , 第(3)空填写:  $S + K$ ; 最后判断N和S是否相等, 第(4)空填写: S。

496的真因子有: 1、2、4、8、16、31、62、124、248,  
 $1 + 2 + 4 + 8 + 16 + 31 + 62 + 124 + 248 = 496$ ; 因此496是完美数, 同时  $496 = (1 + 2 + 3 + 4 + \dots + 30 + 31)$ , 因此496是完美数和三角形数。

第2题: 阅读以下说明和C代码, 填写程序中的空(1)~(5), 将解答写入答题纸的对应栏内。

【说明】

直接插入排序是一种简单的排序方法, 具体做法是: 在插入第i个关键码时,  $k_1, k_2, \dots, k_{i-1}$ 已经排好序, 这时将关键码 $k_i$ 依次与关键码 $k_{i-1}, k_{i-2}, \dots$ , 进行比较, 找到 $k_i$ 应该插入的位置时停下来, 将插入位置及其后的关键码依次向后移动, 然后插入 $k_i$ 。

例如, 对{17, 392, 68, 36}按升序作直接插入排序时, 过程如下:

第1次: 将392 ( $i = 1$ ) 插入有序子序列{17}, 得到{17, 392};

第2次: 将68 ( $i = 2$ ) 插入有序子序列{17, 392}, 得到{17, 68, 392};

第3次: 将36 ( $i = 3$ ) 插入有序子序列{17, 68, 392}, 得到{17, 36, 68, 392}, 完成排序。

下面函数 insertSort 用直接插入排序对整数序列进行升序排列, 在main函数中调用insertSort并输出排序结果。

【C代码】

```
void insertSort(int data[], int n)
/*用直接插入排序法将data[0]~data[n-1]中的n个整数进行升序排列*/
{
    int i, j;
    int tmp;

    for(i = 1; i < n; i++){
        if(data[i] < data[i-1]){ //将data[i]插入有序子序列data[0]~data[i-1]
            tmp = data[i]; //备份待插入的元素
            data[i] = (1);
            for(j = i-2; j >= 0 && data[j] > tmp; j--) //查找插入位置并将元素后移
                (2);
            (3) = tmp; //插入正确位置
        }
    }
}

int main()
{
    int *bp, *ep;
    int n, arr[] = {17, 392, 68, 36, 291, 776, 843, 255};
    n = sizeof(arr) / sizeof(int);
    insertSort(arr, n);
    bp = (4); ep = arr + n;
    for(; bp < ep; bp++) //按升序输出数组元素
        printf("%d ", (5));
    return 0;
}
```

阅读以下说明和C代码, 填写程序中的空(1)~(5), 将解答写入答题纸的对应栏

内。

### 【说明】

直接插入排序是一种简单的排序方法, 具体做法是: 在插入第 $i$ 个关键码时,  $k_1, k_2, \dots, k_{i-1}$ 已经排好序, 这时将关键码 $k_i$ 依次与关键码 $k_{i-1}, k_{i-2}, \dots$ , 进行比较, 找到 $k_i$ 应该插入的位置时停下来, 将插入位置及其后的关键码依次向后移动, 然后插入 $k_i$ 。

例如, 对 { 17, 392, 68, 36 } 按升序作直接插入排序时, 过程如下:

第1次: 将392 ( $i=1$ ) 插入有序子序列{17}, 得到{17, 392};

第2次: 将68 ( $i=2$ ) 插入有序子序列{17, 392}, 得到{17, 68, 392};

第3次: 将36 ( $i=3$ ) 插入有序子序列{17, 68, 392}, 得到{17, 36, 68, 392}, 完成排序。

下面函数 insertSort 用直接插入排序对整数序列进行升序排列, 在 main 函数中调用 insertSort 并输出排序结果。

### 【C代码】

```
void insertSort(int data[],int n)
/*用直接插入排序法将data[0]~ data[n-1]中的n个整数进行升序排列*/
{   int i,j;   int tmp;

    for(i=1;i<n;i++){
        if(data[i]< data[i-1]) { //将data[i]插入有序子序列data[0]~ data[i-1]
            tmp=data[i];       //备份待插入的元素
            data[i]= (1);
            for(j=i-2;j>=0 && data[j] > tmp;j--){ //查找插入位置并将元素后移
                (2);
            }
            (3) =tmp; //插入正确位置
        }
    }
}

int main()
{   int *bp,*ep;
    int n,arr[]={17,392,68,36,291,776,843,255};
    n = sizeof(arr) / sizeof(int);
    insertSort(arr,n);
    bp= (4) ;   ep = arr+n;
    for( ;bp<ep; bp++) //按升序输出数组元素
        printf("%d ", (5) );
    return 0;
}
```

题: 2.1

答案解析:

### 【参考答案】

- (1) data[i-1]
- (2) data[j+1]=data[j]
- (3) data[j+1]
- (4) arr
- (5) \*bp

### 【解析】

直接插入排序法是将关键码插入已经排好的序列中, 因此将data[i]插入序列data[0]~data[i-1]中, 此时序列data[0]~data[i-1]已经按照升序排列好, 而data[i]应插入位置前的数据应该比data[i]小, 而插入位置后的数据应比data[i]大, 在if语句中判断data[i] < data[i-1]中可以看出, 在进行插入运算时, 是从序列data[0]~data[i-1]最后一个数据data[i-1]向前逐一进行比较, 若data[i] >= data[i-1], 则将data[i]插入到data[i-1]后; 若data[i] < data[i-1], data[i]需要与data[i-2]进行比较, 如此依次进行, 此时需要将data[i]备份并将data[i-1]后移, 即temp=data[i]; data[i]=data[i-1]; 之后是进行比较, 即for(j=i-2; j>=0 && data[j]>tmp; j--)循环, 从data[i-2]开始向前逐一比较, 即j从i-2开始向0循环, 若data[j]>tmp, 则进行for循环, 此时需要将data[j]即data[i-2]的值后移, 使得data[i-1]=data[i-2], 即data[j+1]=data[j], 然后j--, 用tmp与data[j]进行比较, 如果tmp < data[j], 则说明tmp应放在data[j]之前, 那么data[j]需要继续往后移动。所以data[j+1]=data[j]。

当该循环结束时, 此时有2种情况: (1) j=-1 < 0, 此时data[0]>tmp; 应使得data[0]后移, 即data[1]=data[0], data[0]=tmp, 因此第3空填写data[j+1]; (2) data[j]<=tmp; 此时需要将tmp插入到data[j]后, 即data[j+1]=tmp。

在main函数中调用insertSort函数并输出数组元素, 在for(; bp<ep; bp++)中循环变量是bp, 因此输出的是bp指向的数组元素, 即调用insertSort函数后返回的数组arr, 因此bp=arr (bp是指针变量, 数组名arr可以直接将数组地址传递给bp); 在printf函数中输出bp; 因此printf("%d", \*bp)。

第3题: 阅读以下C代码, 回答问题1至问题3, 将解答填入答题纸的对应栏内。

### 【C代码1】

```
float adjustSalary(int service_year, int age, float salary) {
    if( service_year <= 5 ) {
        if( age > 30 )
            salary *= 1.2;
    }
    else
        salary *= 2.0;
    return salary;
}
```

### 【C代码2】

```
void foo(int coin) {
    switch (coin) {
        case 1: printf("Cent ");
        case 5: printf("Nickel "); break;
        case 10: printf("Dime ");
        case 25: printf("Quarter ");
    }
}
```

### 【C代码3】

```
int getSum(int n)
{
    int t, i = 0, sum = 0;
    while ( i < n ) {
        scanf("%d", &t);
        if( t<0 )
            continue;
        sum += t;
        i++;
    }
    return sum;
}
```

题 : 3.1 【问题1】 ( 3分 )

对于【C代码1】, 写出下面的函数调用后x1、x2和x3的值。

```
x1 = adjustSalary ( 1,25,9000.0 );
x2 = adjustSalary ( 8,32,7000.0 );
x3 = adjustSalary ( 5,41,5000.0 );
```

题 : 3.2 【问题2】 ( 6分 )

- ( 1 ) 写出函数调用为foo(1)的输出结果 ;
- ( 2 ) 写出函数调用为foo(5)的输出结果 ;
- ( 3 ) 写出函数调用为foo(10)的输出结果 ;
- ( 4 ) 写出函数调用为foo(25)的输出结果。

题 : 3.3 【问题3】 ( 6分 )

- ( 1 ) 简要说明【C代码3】中的函数getSum()的功能 ;
- ( 2 ) 程序员小王将函数getSum改写后得到下面的函数getSum\_v2 ( 即用for语句来描述循环 )。请问, getSum\_v2的功能是否与【C代码3】的getSum完全相同, 若不同, 请说明原因。

答案解析 :

#### 【参考答案】

- ( 1 ) x1=9000.000000
- ( 2 ) x2=14000.000000
- ( 3 ) x3=6000.000000

#### 【解析】

对于x1, service\_year=1, age=25, salary=9000.0, 首先判断service\_year<=5, 因此再判断age<30, 不进行任何运算, salary仍为9000.0, 由于salary为float类型数据, 因此输出为9000.000000。

对于x2, service\_year=8, age=32, salary=7000.0, 首先判断service\_year>5, 因此直接进行else中的运算, salary=7000.0\*2.0=14000.0, 由于salary为float类型数据, 因此输出为14000.000000。

对于x3, service\_year=5, age=41, salary=5000.0, 首先判断service\_year<=5, 因此再判断age>30, 进行运算salary=5000.0\*1.2=6000.0, 由于salary为float类型数据, 因此输出为6000.000000。

答案解析：

【参考答案】

- ( 1 ) foo(1) : Cent  
Nickel
- ( 2 ) foo(5) : Nickel
- ( 3 ) foo(10) : Dime  
Quarter
- ( 4 ) foo(25) : Quarter

【解析】

foo(1) : coin=1, 执行printf( "Cent " ), 输出Cent并回车, 继续执行printf( "Nickel " ), 输出Nickel并回车, 再执行break, 结束foo函数。  
foo(5) : coin=5, 执行printf( "Nickel " ), 输出Nickel并回车, 再执行break, 结束foo函数。  
foo(10) : coin=10, 执行printf( "Dime " ), 输出Dime并回车, 继续执行printf( "Quarter " ), 输出Quarter并回车, 结束foo函数。  
foo(25) : coin=25, 执行printf( "Quarter " ), 输出Quarter并回车, 结束foo函数。

答案解析：

【参考答案】

- ( 1 ) 函数getSum()是求n个不小于0的整数之和。
- ( 2 ) 函数getSum\_v2的功能与函数getSum不同, getSum()是求n个不小于0的整数之和, 计算和的数目总数是n个, 而getSum\_v2是总共输入了n个数, 这n个数种有大于等于0, 也有小于0的数, 最终只把大于等于0的数进行求和, 因此最终计算的和的个数有可能是小于n的。

【解析】

- ( 1 ) 函数getSum()是求n个不小于0的整数之和。
- ( 2 ) 函数getSum\_v2的功能与函数getSum不同, 原因: 在getSum中, 当t<0时, 继续运行scanf函数获得t, 而不会运行i++运算, i的值不会改变, 最终会获得n个不小于0的整数并求和; 而在getSum\_v2中, 当t<0时, 首先进行i++, 再继续scanf函数, 因此当输入的数出现负值时, 会导致最终运算的值可能少于n个。

第4题：阅读以下说明和C代码，填写代码中的空（1）～（6），将解答写入答题纸的对应栏内。

【说明】

下面的C代码在输入的100个英文单词中找出最小单词和最大单词。约定每个单词是仅由英文字母

构成的字符串, 且都不超过20个字符。单词的大小按照字典序定义。例如, 单词 "entry" 大于 "enter"、"art" 小于 "article"、"an" 等于 "An"。

### 【C代码】

```
#include <stdio.h>
#define NUMBER 100
int isValid(const char *s1);           //若字符串s1仅包含英文字母则返回1, 否则返回0
char toLower(char ch);                 //将大写字母转换为小写字母
int usr_strcmp(char *s1, char *s2);    //比较字符串s1和s2, 相等时返回0,
                                        //s1大则返回正整数, s1小则返回负整数
void usr_strcpy(char *s1, const char *s2); //字符串s2拷贝给s1

int main()
{
    char word[32];
    char maxWord[32] = "", minWord[32] = "";
    int numWord = 0;
    while(numWord < NUMBER) {
        scanf("%s", (1) );           //输入一个单词存入word
        if(isValid(word)) {
            if (0 == numWord) {usr_strcpy(minWord, word); usr_strcpy(maxWord, word);}
            numWord++;
            if( (2) > 0)               //调用usr_strcmp比较单词
                usr_strcpy(maxWord, word); //用maxWord记下最大单词
            else
                if( (3) < 0)           //调用usr_strcmp比较单词
                    usr_strcpy(minWord, word); //用minWord记下最小单词
        }
    }
    printf("max Word=%s min Word=%s ", maxWord, minWord);
    return 0;
}

int isValid(const char *s)
{
    for( *s; *s; s++)
        if(!(*s >= "a" && *s <= "z") && !(*s >= "A" && *s <= "Z"))
            return 0;
    return 1;
}

char toLower(char ch)
{
    //若ch为大写字母则返回其小写形式, 否则直接返回原字符
    if(ch >= "A" && ch <= "Z")
        ch = (4) + "a";
    return ch;
}

int usr_strcmp(char *s1, char *s2)
{
    //按字典序比较两个英文单词, 若s1表示的单词大, 则返回正整数,
    //若s1表示的单词小, 则返回负整数; 否则返回0
}
```



```
for( ( 5 ) ); {  
    if(toLower(*s1)==toLower(*s2)) {s1++,s2++;}  
    else  
        break;  
}  
return(toLower(*s1) - toLower(*s2));  
}
```

```
void usr_strcpy(char *s1,const char *s2)  
{ //将s2表示的字符串复制给s1  
    for( ( 6 ) );  
        *s1++= *s2++;  
    *s1= ' ' ;  
}
```

题 : 4.1

答案解析 :

【参照答案】

- ( 1 ) word
- ( 2 ) usr\_strcmp(word, maxWord)
- ( 3 ) usr\_strcmp(word, minWord)
- ( 4 ) ch- 'A'
- ( 5 ) \*s1!=' ' &&\*s2!=' ' 或\*s1||\*s2等价表示
- ( 6 ) \*s2!=' ' 或\*s2等价表示

【解析】

( 1 ) scanf函数是将输入的数据赋值给相应的变量, 在格式字符串后需要取变量地址, 即将数据赋值给相应地址的变量, word是数组名, 因此可以直接作为地址。

( 2 ) 调用usr\_strcmp比较单词, 当usr\_strcmp返回的值为正整数时, 调用usr\_strcpy将word值赋值给maxWord, 因此此处是比较word和maxWord, 即usr\_strcmp(word, maxWord)

( 3 ) 调用usr\_strcmp比较单词, 当usr\_strcmp返回的值为负整数时, 调用usr\_strcpy将word值赋值给minWord, 因此此处是比较word和minWord, 即usr\_strcmp(word, minWord)

( 4 ) toLower函数是将大写字母转换为小写字母, if函数是判断ch是否为大写字母, 当ch为大写的时候, 将ch转换为小写, 小写是在' a' 的基础上加上一个值得到, 例如 'b' =1+' a' , 这个值便是大写字母和 'A' 之间的差值, 即例子中1为' B' - 'A' , 因此此处填写ch- 'A'

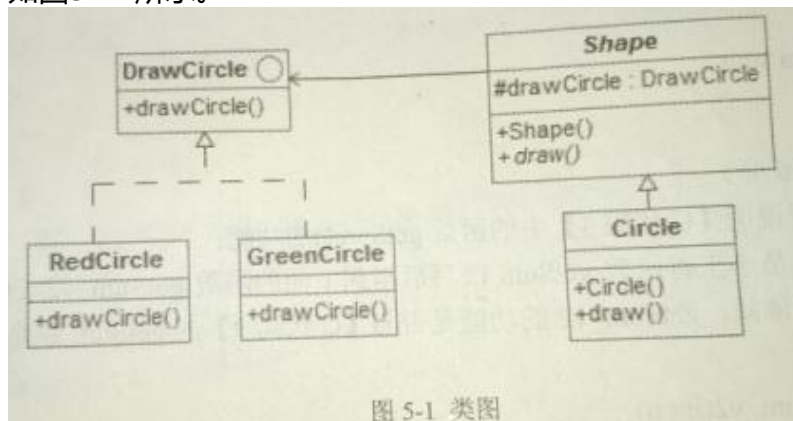
( 5 ) usr\_strcmp是比较两个英文单词, 是需要对s1和s2中的字符逐一比较, 因此在for循环中进行比较, for循环的判断条件是s1和s2中的字符是否为' ' , 因此此处填写\*s1!=' ' &&\*s2!=' '

( 6 ) usr\_strcpy是将s2复制给s1, 是需要将s2中的字符逐一赋值给s1, 因此在for循环中进行赋值, for循环的判断条件是s2中的字符是否为' ' , 因此此处填写\*s2!=' '

第5题：阅读以下说明和Java程序，填写程序中的空（1）~（6），将解答写入答题纸的对应栏内。

【说明】

以下Java代码实现一个简单绘图工具，绘制不同形状以及不同颜色的图形。部分接口、类及其关系如图5 - 1所示。



【Java代码】

```
interface DrawCircle { //绘制圆形
    public (1) ;
}

class RedCircle implements DrawCircle { //绘制红色圆形
    public void drawCircle(int radius,int x, int y) {
        System.out.println("Drawing Circle[red,radius:" + radius + ",x:" + x + ",y:" + y+ "]);
    }
}

class GreenCircle implements DrawCircle { //绘制绿色圆形
    public void drawCircle(int radius, int x, int y) {
        System.out.println("Drawing Circle[green,radius:" +radius+ ",x: " +x+ ",y: " +y+ "]);
    }
}

abstract class Shape { //形状
    protected (2) ;

    public Shape(DrawCircle drawCircle) {
        this.drawCircle = drawCircle;
    }
    public abstract void draw();
}

class Circle extends Shape { //圆形
    private int x,y,radius;

    public Circle(int x,int y,int radius,DrawCircle drawCircle) {
        (3) ;
        this.x = x;
        this.y = y;
        this.radius = radius;
    }
}
```

```
}

public void draw() {
    drawCircle.  (4) ;
}

}

public class DrawCircleMain {
    public static void main(String[] args) {
        Shape redCircle = new Circle( 100,100,10,  (5) ); //绘制红色圆形
        Shape greenCircle = new Circle(200,200,10,  (6) ); //绘制绿色圆形

        redCircle.draw();
        greenCircle.draw();
    }
}
```

题 : 5.1

答案解析 :

【参考答案】

- ( 1 ) void drawCircle (int radius,int x,int y)
- ( 2 ) DrawCircle drawCircle
- ( 3 ) super.drawcircle=drawcircle
- ( 4 ) drawCircle(radius,x,y)
- ( 5 ) new redCircle
- ( 6 ) new greenCircle

【解析】

第一空是填接口里面的方法, 在接口的实现里面找, 可以发现应该填void drawCircle (int radius,int x,int y)。

第二空可以根据后面this drawCircle=drawCircle判断, 这里应该有一个drawCircle属性, 因此应该填) DrawCircle drawCircle。

第三空这里用super, 用super. drawcircle来引用父类的成员。

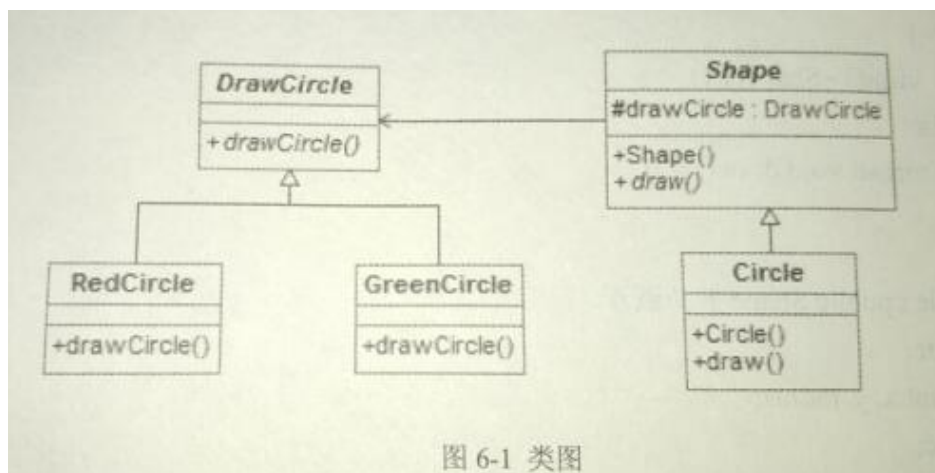
第四空调用drawCircle(radius,x,y)方法。

第五、六空分别创建一个红色圆形对象和一个绿色圆形对象作为Circle里面的实参。

第6题: 阅读下列说明和C++代码, 填写程序中的空(1)~(6), 将解答写入答题纸的对应栏内。

### 【说明】

以下C++代码实现一个简单绘图工具, 绘制不同形状以及不同颜色的图形。部分类及其关系如图6-1所示。



### 【C++代码】

```
#include <iostream>
#include <string>
using namespace std;
class DrawCircle {    //绘制圆形, 抽象类
public:
    (1) ; //定义参数为 int radius, int x, int y
    virtual ~DrawCircle() {}
};

class RedCircle : public DrawCircle { //绘制红色圆形
public:
    void drawCircle(int radius, int x, int y) {
        cout << "Drawing Circle[red,radius: " << radius;
        cout << ",x: " << x << ",y: " << y << "]" << endl;
    }
};

class GreenCircle : public DrawCircle { //绘制绿色圆形
public:
    void drawCircle(int radius, int x, int y) {
        cout << "Drawing Circle[green,radius: " << radius;
        cout << ",x: " << x << ",y: " << y << "]" << endl;
    }
};

class Shape { //形状, 抽象类
protected:
    (2) ;
public:
    Shape(DrawCircle *drawCircle) {
        this -> drawCircle = drawCircle;
    }
};
```

```
    }  
    virtual ~shape() { }  
public:  
    virtual void draw() = 0;  
};  
  
class Circle : public Shape { //圆形  
private:  
    int x,y,radius;  
public:  
    Circle(int x,int y,int radius,DrawCircle *drawCircle)    (3)    {  
        this->x = x;  
        this->y = y;  
        this->radius = radius;  
    }  
public:  
    void draw() {  
        drawCircle ->    (4)    ;  
    }  
};  
  
int main()  
{  
    Shape *redCircle = new Circle(100,100,10,    (5)    );//绘制红色圆形  
    Shape *greenCircle = new Circle(100,100,10,    (6)    );//绘制绿色圆形  
    redCircle ->draw();  
    greenCircle ->draw();  
    return 0;  
}
```

题 : 6.1

答案解析 :

【参考答案】

- ( 1 ) void drawCircle (int radius,int x,int y)
- ( 2 ) DrawCircle\*drawCircle
- ( 3 ) drawcircle
- ( 4 ) drawCircle(radius,x,y)
- ( 5 ) new redCircle()
- ( 6 ) new greenCircle()

【解析】

第一空是填接口里面的方法，在接口的实现里面找，可以发现应该填void drawCircle (int radius,int x,int y)。

第二空可以根据后面this drawCircle=drawCircle判断，这里应该有一个drawCircle属性，因此应该填 ) DrawCircle drawCircle。

第三空这里填drawcircle, 用-> drawcircle来引用父类的成员。

第四空调用drawCircle(radius,x,y)方法。

第五、六空分别创建一个红色圆形对象和一个绿色圆形对象作为Circle里面的实参。

跨步软考 (kuabu.xyz)  
手机APP, 微信公众号