

全国计算机技术与软件专业技术资格（水平）考试

2007 年下半年 程序员 上午试卷

（考试时间 9：00～11：30 共 150 分钟）

请按下述要求正确填写答题卡


1. 在答题卡的指定位置上正确写入你的姓名和准考证号，并用正规 2B 铅笔在你写入的准考证号下填涂准考证号。
2. 本试卷的试题中共有 75 个空格，需要全部解答，每个空格 1 分，满分 75 分。
3. 每个空格对应一个序号，有 A、B、C、D 四个选项，请选择一个最恰当的选项作为解答，在答题卡相应序号下填涂该选项。
4. 解答前务必阅读例题和答题卡上的例题填涂样式及填涂注意事项。解答时用正规 2B 铅笔正确填涂选项，如需修改，请用橡皮擦干净，否则会导致不能正确评分。

例题

● 2007 年下半年全国计算机技术与软件专业技术资格（水平）考试日期是
（88）月（89）日。

- | | | | |
|-----------|-------|-------|------|
| （88）A. 12 | B. 11 | C. 10 | D. 9 |
| （89）A. 6 | B. 5 | C. 4 | D. 3 |

因为考试日期是“11 月 3 日”，故（88）选 B，（89）选 D，应在答题卡序号 88 下对 B 填涂，在序号 89 下对 D 填涂（参看答题卡）。

● 在 Word 编辑状态下,有些英文单词和汉字下面会自动加上红色或绿色的波浪型细下划线。以下叙述中,“波浪型细下划线_(1)_”是错误的;按_(2)_键与工具栏上的按钮功能相同。









- (1) A. 提醒用户此处可能有拼写或语法错误
 B. 为红色表示可能是拼写问题,为绿色可能是语法问题
 C. 是文档内容的一部分,打印文档时会被打印出来
 D. 不属于文档内容,打印文档时不会被打印出来

- (2) A. Ctrl+C B. Ctrl+V C. Ctrl+A D. Ctrl+S

● 在 Excel 中,通过冻结或者拆分窗格可以在滚动工作表时始终保持部分数据可见。下图中_(3)_,当鼠标指针在上述位置变为_(4)_后,将该框拖至所需的位置即可。



- (3) A. ①和③分别表示水平拆分框和垂直冻结框
 B. ①和③分别表示水平冻结框和垂直冻结框
 C. ②和③分别表示垂直冻结框和水平拆分框
 D. ②和③分别表示水平拆分框和垂直拆分框

- (4) A. 或 B. 或 C. 或 D. 或

● 给定 URL 为 <http://www.ceiaec.org/index.htm>,其中 www.ceiaec.org 表示_(5)_____。

- (5) A. 所用的协议 B. 所访问网站的域名
 C. 请求查看的文档名 D. 电子邮件地址

● 计算机中数据输入输出的控制方式有多种,“中断”方式的优点不包括_(6)_____。

- (6) A. I/O 与 CPU 并行处理 B. 并行处理多种 I/O
 C. 实时响应突发事件 D. 批量传送数据

● 欲知 8 位二进制数($b_7b_6b_5b_4b_3b_2b_1b_0$)的 b_2 是否为 1,可将该数与二进制数 00000100 进行_(7)_____运算,若运算结果不为 0,则此数的 b_2 必为 1。

- (7) A. 加 B. 减 C. 与 D. 或

● 光盘驱动器与主机的接口总线常采用 (8) 总线。

- (8) A. ISA B. CETRONIC
C. EIDE (ATA) D. PCI

● 汉字机内码与国标码的关系为：机内码 = 国标码 + 8080H。若已知某汉字的国标码为 3456H，则其机内码为 (9) 。

- (9) A. B4D6H B. B536H C. D4B6H D. C4B3H

● 设某系统由 P1 和 P2 两个子系统组成，当且仅当两个子系统都能正常工作时，该系统才能正常工作。若 P1 和 P2 的可靠性均为 0.9，则该系统的可靠性是 (10) 。

- (10) A. 0.1 B. 0.45 C. 0.81 D. 0.9

● 评价计算机系统性能时，MIPS 是衡量 (11) 的一种单位。

- (11) A. 时钟频率 B. 运算速度
C. 系统可靠性 D. 系统失效率

● 下面的图像格式中，（12）不支持透明控制信息。

- (12) A. GIF B. PNG C. PSD D. JPG

● 对同一段音乐可以选用 MIDI 格式或 WAV 格式来记录存储。以下叙述中, (13) 是不正确的。

- (13) A. WAV 格式的音乐数据量比 MIDI 格式的音乐数据量大
B. 记录演唱会实况不能采用 MIDI 格式的音乐数据
C. WAV 格式的音乐数据没有体现音乐的曲谱信息
D. WAV 格式的音乐数据和 MIDI 格式的音乐数据都能记录音乐波形信息

● 某块 SoundBlaster 声卡是 8 位的，这里的“8 位”指 (14)。

- (14) A. 声音采集最大量化精度是 8 位
B. MIDI 通道数是 8
C. 可以产生的音色数是 2^8
D. 声道数为 8

● 使用 IE 浏览器浏览网页时，出于安全方面的考虑，需要禁止执行 Java Script，可以在 IE 中 （15） 。

- (15) A. 禁用 ActiveX 控件 B. 禁用脚本
C. 禁用没有标记为安全的 ActiveX 控件 D. 禁用 cookie

● 在公司内网中部署 (16) 可以最大限度防范内部攻击。

- (16) A. 防火墙
B. 电磁泄密及防护系统
C. 邮件过滤系统
D. 入侵检测系统

● 向商标局提出注册商标转让申请的人应当是（17）。

(17) A. 转让人 B. 受让人 C. 商标注册人 D. 转让人和受让人

● 下列标准代号中，（18）表示国际标准。

(18) A. GJB B. IEC C. GB/T D. ANSI

● 用补码表示的 8 位二进制数 11100000，其值为十进制数（19）。

(19) A. -31 B. -32 C. -64 D. -65

● 用 ASCII 码表示的大写英文字母 B（42H）加偶校验后的二进制编码为（20）。

(20) A. 10001000 B. 10000010 C. 11000001 D. 01000010

● CPU 主要由运算单元（ALU）、控制单元（CU）、寄存器和时序电路组成，对指令进行译码的功能是由（21）实现的。

(21) A. 运算单元 B. 控制单元 C. 寄存器 D. 时序电路

● 两个带符号的数进行运算时，在（22）的情况下有可能产生溢出。

(22) A. 同符号数相加 B. 同符号数相减
C. 异符号数相加 D. 异符号数相“或”

● 主板（也称母板或系统板）是计算机硬件系统集中管理的核心载体，几乎集中了全部系统功能，是计算机中的重要部件之一。下图所示的微机主板上，①处是（23），②处是（24）。



(23) A. CPU 插槽 B. 内存插槽 C. PCI 插槽 D. IDE 插槽

(24) A. CPU 插槽 B. 内存插槽 C. PCI 插槽 D. IDE 插槽

● 计算机加电自检以后，引导程序首先装入(25)，否则，计算机不能做任何事情。

(25) A. 操作系统 B. 编译程序 C. Office 系列软件 D. 应用软件

● 已知有 10 个进程共享一个互斥段，如果最多允许 6 个进程同时进入互斥段，则信号量 S 的初值为(26)，信号量 S 的变化范围是(27)。

(26) A. 0 B. 1 C. 5 D. 6

(27) A. 0~10 B. -1~9 C. -4~6 D. -5~5

● (28) 不属于程序语言翻译软件。

(28) A. 编译程序 B. 解释程序 C. 汇编程序 D. 编辑程序

● 若程序 P 经编译并链接后可执行，则(29)。

(29) A. P 是正确的程序 B. P 中没有语法错误
C. P 中没有逻辑错误 D. P 在运行中不会出错

● 若程序中定义了局部变量与全局变量，则(30)。

(30) A. 在函数内只能访问局部变量，但不能访问全局变量
B. 局部变量与全局变量的名字不得相同
C. 一个函数被多次调用时，其定义的局部变量必定分配同一个存储单元
D. 函数中定义的局部变量与全局变量同名时，在函数内引用的是局部变量

● (31) 不是良好的编程风格。

(31) A. 变量名采用单字母符号或单字母加数字串
B. 程序代码采用缩进格式
C. 为变量设置初始值
D. 程序中包含必要的注释

● 函数调用采用“(32)”方式时，系统将实参的地址传递给形式参数。

(32) A. 传值调用 B. 引用调用 C. 宏调用 D. 内部调用

● 若程序中使用的变量未设置初始值，则(33)。

(33) A. 编译时一定出错 B. 运行时一定会出错
C. 链接时一定出错 D. 运行结果可能出错

● 在统一建模语言 (Unified Modeling Language, UML) 中, 描述本系统与外部系统及用户之间交互的图是 (34); 按时间顺序描述对象间交互的图是 (35)。

(34) A. 用例图 B. 类图 C. 对象图 D. 状态图

(35) A. 部署图 B. 序列图 C. 协作图 D. 状态图

● n 个元素依次全部进入栈后, 再陆续出栈并经过一个队列输出。那么, (36)。

(36) A. 元素的出队次序与进栈次序相同

B. 元素的出队次序与进栈次序相反

C. 元素的进栈次序与进队次序相同

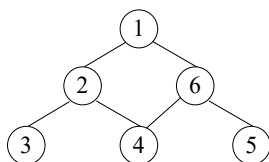
D. 元素的出栈次序与出队次序相反

● 若一个栈以向量 $V[1..n]$ 存储, 且空栈的栈顶指针 top 为 $n+1$, 则将元素 x 入栈的正确操作是 (37)。

(37) A. $top = top+1; V[top] = x;$ B. $V[top] = x; top = top+1;$

C. $top = top-1; V[top] = x;$ D. $V[top] = x; top = top-1;$

● 广度优先遍历的含义是: 从图中某个顶点 v 出发, 在访问了 v 之后依次访问 v 的各个未被访问过的邻接点, 然后分别从这些邻接点出发依次访问它们的邻接点, 且“先被访问的顶点的邻接点”先于“后被访问的顶点的邻接点”被访问, 直至图中所有已被访问的顶点的邻接点都被访问到。 (38) 是下图的广度优先遍历序列。



(38) A. 1 2 6 3 4 5 B. 1 2 3 4 5 6 C. 1 6 5 2 3 4 D. 1 6 4 5 2 3

● 对于长度为 11 的顺序存储的有序表, 若采用折半查找 (向下取整), 则找到第 5 个元素需要与表中的 (39) 个元素进行比较操作 (包括与第 5 个元素的比较)。

(39) A. 5 B. 4 C. 3 D. 2

● 与单向链表相比, 双向链表 (40)。

(40) A. 需要较少的存储空间

B. 遍历元素需要的时间较短

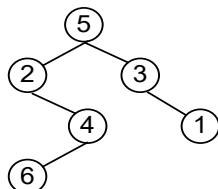
C. 较易于访问相邻结点

D. 较易于插入和删除元素

● 如果待排序序列中两个元素具有相同的值，在排序前后它们的相互位置发生颠倒，则称该排序算法是不稳定的。(41)是稳定的排序方法，因为这种方法在比较相邻元素时，值相同的元素并不进行交换。

(41) A. 冒泡排序 B. 希尔排序 C. 快速排序 D. 简单选择排序

● 对下图所示的二叉树进行中序遍历（左子树、根、右子树）的结果是(42)。



(42) A. 2 5 3 4 6 1

B. 2 5 3 4 1 6

C. 2 6 5 4 1 3

D. 2 6 4 5 3 1

● 采用一维数组 S 存储一个 n 阶对称矩阵 A 的下三角部分（按行存放，包括主对角线），设元素 A[i][j] 存放在 S[k] 中（i、j、k 均从 1 开始取值），且 S[1]=A[1][1]，则 k 与 i、j 的对应关系是(43)。例如，元素 A[3][2] 存在 S[5] 中。

(43) A. $k = \frac{i(i+1)}{2} + j - 1$

B. $k = \frac{i(i+1)}{2} + j$

C. $k = \frac{i(i-1)}{2} + j - 1$

D. $k = \frac{i(i-1)}{2} + j$

● 关于对象和类的叙述，正确的是(44)。

(44) A. 如果两个对象的所有成员变量的值相同，则这两个对象是同一对象

B. 编写代码时，不允许使用匿名类

C. 不同的对象一定属于不同的类

D. 每个对象都有惟一标识，以彼此区分

● (45)关系描述了某对象由其他对象组成。

(45) A. 依赖

B. 一般化

C. 聚合

D. 具体化

● 关于对象封装的叙述，正确的是(46)。

(46) A. 每个程序模块中都封装了若干个对象

B. 封装可实现信息隐藏

C. 封装使对象的状态不能改变

D. 封装是把对象放在同一个集合中

● 在面向对象技术中，抽象类是（47）。

- (47) A. 接口
B. 没有方法的类
C. 没有子类的类
D. 至少有一个未定义消息的类

● 在面向对象技术中，多态性的含义是（48）。

- (48) A. 一个对象可同时指向不同的类的对象
B. 同一消息在不同时刻可调用不同的方法
C. 对象的生存周期中包含对象的多种状态
D. 不同消息可以调用相同的方法

● (49) 语言最适用于过程式程序设计。

- (49) A. C B. HTML C. JAVA D. PROLOG

● (50) 是正确的叙述。

- (50) A. 解释方式比编译方式更适用于规模较大的程序
B. 解释方式下生成的目标代码长, 编译方式下生成的目标代码短
C. 解释方式下不生成源程序的目标代码
D. 解释方式只能用于批处理

● 由多个源文件组成的 C 程序，经过编辑、预处理、编译、链接等阶段才能生成最终的可执行程序。此过程中，在（51）阶段可以发现被调用的函数未定义。

- (51) A. 编辑和预处理 B. 预处理 C. 编译 D. 链接

● 在软件开发过程中, (52)。

- (52) A. 增加更多的程序员可以加快软件开发速度
B. 只有对目标的一般描述还不能立即开始编码
C. 若项目需求有变化, 应先在编码中进行处理以不耽误开发进度
D. 程序的质量只能在用户使用之后才能进行评估

● 采用结构化方法开发软件时，常使用数据流图来描述系统数据处理过程，它是（53）阶段产生的。

- (53) A. 系统分析 B. 概要设计 C. 详细设计 D. 编码

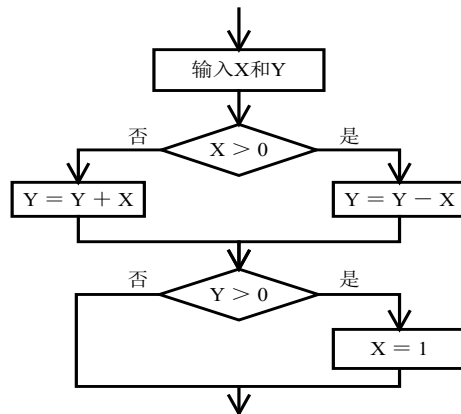
● 软件维护可分为改正性维护、适应性维护、完善性维护和预防性维护。在软件的整个维护活动中, (54) 维护所占比重最大。

- (54) A. 改正性 B. 适应性 C. 完善性 D. 预防性

● 在软件的使用过程中，用户往往会对软件提出新的功能要求与性能要求。为了满足这些要求而进行的维护属于(55)维护。

- (55) A. 改正性 B. 适应性 C. 完善性 D. 预防性

● 对于如下的程序流程，若采用语句覆盖法设计测试案例（使每条语句都至少执行 1 次），则至少需要设计(56)个测试案例。



- (56) A. 1 B. 2 C. 3 D. 4

● 关系代数运算是以集合操作为基础的运算，其五种基本运算是并、差、(57)、投影和选择，其他运算可由这些运算导出。为了提高数据的操作效率和存储空间的利用率，需要对(58)进行分解。

- (57) A. 交 B. 连接 C. 笛卡尔积 D. 自然连接
(58) A. 内模式 B. 视图 C. 外模式 D. 关系模式

● 设有关系 Students（学号，姓名，年龄，性别，系名，家庭住址），其中，属性“系名”是关系 DEPT 的主键，用户要求访问“家庭住址”的某个成分，如邮编、省、市、街道以及门牌号。关系 Students 的主键和外键分别是(59)。“家庭住址”是一个(60)属性。

- (59) A. 学号、家庭住址 B. 姓名、家庭住址
C. 学号、系名 D. 姓名、系名
(60) A. 简单 B. 复合 C. 多值 D. 派生

● 设有关系 Students（学号，姓名，年龄，性别，系名，家庭住址），如果要查询姓“李”的且家庭住址包含“科技路”的学生学号、姓名以及所在系，则对应的 SELECT 语句如下：

SELECT 学号，姓名，系名

FROM Students

WHERE (61) AND (62);

- (61) A. 姓名 LIKE '李__'
C. 姓名 AS '李__'
(62) A. 家庭住址 LIKE '%科技路%'
C. 家庭住址 AS '%科技路%'
- B. 姓名 LIKE '李%'
D. 姓名 AS '李%'
B. 家庭住址 LIKE '*科技路*'
D. 家庭住址 AS '*科技路*'

● 为了用一个数代表一批数，人们常用这批数据的算术平均值（简称平均值）或中位数来代表。中位数就是位于这批数中间的数（大于它的数与小于它的数一样多）。对于奇数个数而言，排序后很容易确定中间那个数；对于偶数个数而言，排序后中间会有两个数，再取这两个数的算术平均，就是中位数。以下关于平均值与中位数的叙述中，(63)是不正确的。

- (63) A. 中位数比平均值稳健，不易受极端值影响
B. 每个数据加倍后，平均值也加倍；每个数据增加 1 后，平均值也增加 1
C. 三组各 n 个数据有三个中位数，它们的中位数就是这三组数据全体的中位数
D. 三组各 n 个数据有三个平均值，它们的平均值就是这三组数据全体的平均值

● 某村领导需要估计该村某池塘中鱼的大致数量。技术人员想出了一个办法：先从池塘中捕出 30 条鱼，在每条鱼身上做一记号后，又放回池塘。几天后，再从该池塘中捕出 40 条鱼，发现其中有 2 条是有记号的。因此，他估计该池塘大致有(64)条鱼。

类似地，为估计某程序中所含的错误个数，程序员 A 对该程序进行测试时发现了 5 个错误，程序员 B 独立对该程序进行测试时发现了 8 个错误，但其中 2 个错误是程序员 A 已经发现的。因此，人们可以估计，该程序中大致含有(65)个错误。

- (64) A. 300 B. 400 C. 600 D. 1200
(65) A. 12 B. 15 C. 16 D. 20

● 客户端通过(66)协议发送电子邮件。

- (66) A. FTP B. SMTP
C. HTTP D. POP3

● 在 URL 中不填写协议类型，IE 浏览器默认使用(67)协议。

- (67) A. FTP B. Gopher
C. File D. HTTP

● 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(68)。

- (68) A. 网络服务器忙 B. 通信线路忙
C. 本地终端遭遇病毒 D. 没有访问权限

● 在 Windows XP 中, 可以使用____(69)____来浏览日志文件, 其中包含有____(70)____、安全性日志和系统日志。

- (69) A. 事件查看器 B. 浏览器 C. 超级终端 D. 信息服务
(70) A. 用户访问日志 B. 应用程序日志
C. 记帐日志 D. 网络攻击日志

● A ____ (71) ____ is a functional unit that interprets and carries out instructions.

- (71) A. memory B. processor C. storage D. network

● A ____ (72) ____ consists of the symbols, characters, and usage rules that permit people to communicate with computer.

- (72) A. programming language B. network
C. keyboard D. display

● ____ (73) ____ software, also called end-user program, includes database programs, word processors, spreadsheets etc.

- (73) A. Application B. System C. Compiler D. Utility

● In ____ (74) ____, the only element that can be deleted or removed is the one that was inserted most recently.

- (74) A. a line B. a queue C. an array D. a stack

● Most ____ (75) ____ measures involve data encryption and password.

- (75) A. security B. hardware C. display D. program

全国计算机技术与软件专业技术资格（水平）考试

2007 年下半年 程序员 下午试卷

（考试时间 14:00～16:30 共 150 分钟）

请按下述要求正确填写答题纸

1. 在答题纸的指定位置填写你所在的省、自治区、直辖市、计划单列市的名称。
2. 在答题纸的指定位置填写准考证号、出生年月日和姓名。
3. 答题纸上除填写上述内容外只能写解答。
4. 本试卷共 7 道题，试题一至试题四是必答题，试题五至试题七选答 1 道。
每题 15 分，满分 75 分。

试题号	一～四	五～七
选择方法	必答题	选答 1 题

5. 解答时字迹务必清楚，字迹不清时，将不评分。
6. 仿照下面例题，将解答写在答题纸的对应栏内。

例题

2007 年下半年全国计算机技术与软件专业技术资格（水平）考试日期是 (1) 月 (2) 日。

因为正确的解答是“11 月 3 日”，故在答题纸的对应栏内写上“11”和“3”（参看下表）。

例题	解答栏
(1)	11
(2)	3

试题一（共 15 分）

阅读以下说明和流程图，填补流程图中的空缺（1）～（5），将解答填入答题纸的对应栏内。

[说明]

某单位动态收集的数据中常包含重复的数据，所以需要进行处理，使得重复的数据仅出现一次。下面流程图的功能是：在 n ($n \geq 1$) 个数据 D_1 、 D_2 、 \dots 、 D_n 中，选出其中所有不重复的 k 个数据，置于原来前 k 个数据的位置上。

该流程图的算法如下：第 1 个数据必然被选出，然后从第 2 个数据开始，逐个考察其余的数据。假设 D_1 、 D_2 、 \dots 、 D_m ($m \geq 1$) 是已经选出的、不重复的数据，则对于数据 D_i ($m < i \leq n$)，将其依次与 D_m 、 D_{m-1} 、 \dots 、 D_1 进行比较，若没有发现与之相同者，则 D_i 被选出并置于 D_{m+1} 的位置上；否则对 D_i 不做处理。

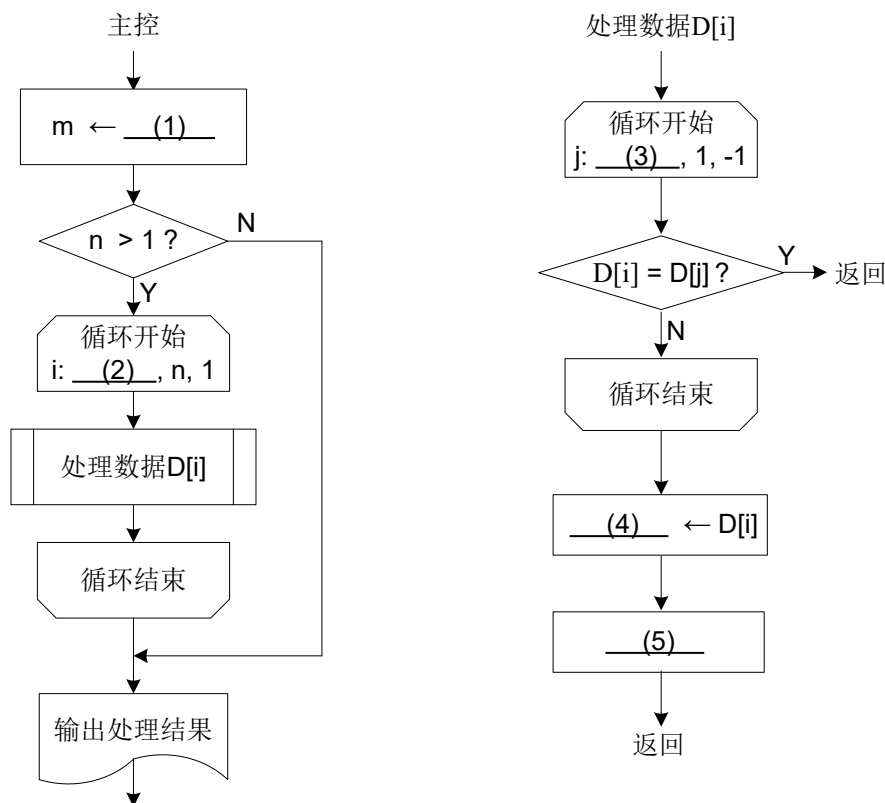
例如，如下 10 个数据：

5, 2, 2, 7, 4, 4, 7, 1, 9, 1 ($n=10$)

经过上述算法处理后的结果为：

5, 2, 7, 4, 1, 9 ($k=6$)

[流程图]



注：循环开始的说明按照“循环变量名：循环初值，循环终值，增量”格式描述。

试题二（共 15 分）

阅读以下说明和 C 语言函数，将应填入 (n) 处的字句写在答题纸的对应栏内。

[说明]

已知 1900 年 1 月 1 日是星期一，下面的函数 count_5_13(int year) 用于计算给定的年份 year 中有几个“黑色星期五”。“黑色星期五”指既是 13 日又是星期五的日期。

函数 count_5_13(int year) 首先算出年份 year 的 1 月 13 日是星期几，然后依次计算每个月的 13 日是星期几，若是星期五，则计数。

程序中使用了函数 isLeapYear(int year)，其功能是判断给定年份是否为闰年，返回值为 1(或 0) 分别表示 year 是(或不是) 闰年。

[C 语言函数]

```
int count_5_13(int year)
{
    int date;          /* date 为 0 表示星期日，为 1~6 分别表示星期一至星期六 */
    long days = 0;     /* days 记录天数 */
    int m, y, c = 0;   /* c 用于表示黑色星期五的个数 */

    if (year < 1900)    return -1;

    /* 计算从 1900 年 1 月 1 日起，至给定年份 year 的 1 月 13 日间隔的天数 */
    days = 12;
    for (y = 1900; y < year; y++) {
        days += 365;
        if (isLeapYear(y)) (1);
    }
    date = ((days % 7) + 1) % 7; /* 算出给定年份 year 的 1 月 13 日是星期几 */
    c = (2) ? 1 : 0;
    for(m = 1; (3); m++) {
        switch (m) {
            case 1: case 3: case 5: case 7: case 8: case 10: case 12:
                days = 31; break;
            case 4: case 6: case 9: case 11:
                days = 30; break;
            case 2: days = 28;
                if ((4))    days = 29;
                break;
        } /* end of switch */
        date = ((days % 7) + (5)) % 7;
        if (date == 5)    c++;
    } /* end of for */
    return c;
}
```

试题三（共 15 分）

阅读以下说明和 C 语言程序，将应填入 (n) 处的字句写在答题纸的对应栏内。

[说明]

某电信公司记录了每个用户的详细通话情况（每次通话数据记录在一行），现将某用户某月的通话数据存入一个文本文件“dial.txt”，其数据格式如下：

拨入或拨出标记 通话开始时间 通话结束时间 对方号码

注 1：数据字段以一个空格作为分隔符。

注 2：拨入和拨出标记均为小写字母。拨入标记为“i”，表示其他用户呼叫本机，本机用户不需付费；拨出标记为“o”，表示本机呼叫其他用户，此时本机用户需要付费。

注 3：通话开始和结束时间的格式均为：HH:MM:SS。其中 HH 表示小时，取值 00~23；MM 表示分钟，取值 00~59；SS 表示秒，取值 00~59。从通话开始到结束这段时间称为通话时间，假定每次通话时间以秒为单位，最短为 1 秒，最长不超过 24 小时。

注 4：跨月的通话记录计入下个月的通话数据文件。

例如“o 23:01:12 00:12:15 …”表示本次通话是本机呼叫其他用户，时间从 23 时 01 分 12 秒至次日的 0 时 12 分 15 秒，通话时间为 71 分 03 秒。

下面程序的功能是计算并输出该用户本月电话费(单位:元)。

通话计费规则为：

1. 月通话费按每次通话费累加；
2. 每次的通话费按通话时间每分钟 0.08 元计算，不足 1 分钟时按 1 分钟计费。

对于每次的拨出通话，程序中先分别计算出通话开始和结束时间相对于当日 0 点 0 分 0 秒的时间长度（以秒为单位），然后算出本次通话时间和通话费。

例如，若输入文件 dial.txt 的数据如下所示，则输出 fee = 7.44。

```
o 14:05:23 14:11:25 82346789
i 15:10:00 16:01:15 13890000000
o 10:53:12 11:07:05 63000123
o 23:01:12 00:12:15 13356789001
```

[C 程序代码]

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

```
FILE *fin;
int main()
{
    char str[80];
    int h1,h2,m1,m2,s1,s2;
    long t_start,t_end, interval;
    int c;
    double fee = 0;
```

```

fin = fopen("dial.txt", "r");
if (!fin)
    return -1;

while (!feof(fin)) {
    if (!fgets(str, 80, fin)) break;

    if (____(1)____) continue;

    h1 = (str[2] - 48) * 10 + str[3] - 48;
    m1 = (str[5] - 48) * 10 + str[6] - 48;
    s1 = (str[8] - 48) * 10 + str[9] - 48;

    h2 = (str[11] - 48) * 10 + str[12] - 48;
    m2 = (str[14] - 48) * 10 + str[15] - 48;
    s2 = (str[17] - 48) * 10 + str[18] - 48;

    t_start = h1*60*60 + m1*60 + s1; /* 通话开始时间 */
    t_end = h2*60*60 + m2*60 + s2; /* 通话结束时间 */

    if (____(2)____) /* 若通话开始和结束时间跨日 */
        interval = ____ (3) ____ - t_start + t_end;
    else
        interval = t_end - t_start;

    c = ____ (4) ____; /* 计算完整分钟数表示的通话时间 */
    if (interval % 60)
        ____ (5) ____;
    fee += c * 0.08;
}
fclose(fin);
printf("fee = %.2lf\n", fee);
return 0;
}

```


试题四（共 15 分）

阅读以下说明和 C 语言函数，将应填入 (n) 处的字句写在答题纸的对应栏内。

[说明]

已知包含头结点（不存储元素）的单链表的元素已经按照非递减方式排序，函数 compress(NODE *head)的功能是去掉其中重复的元素，使得链表中的元素互不相同。

处理过程中，当元素重复出现时，保留元素第一次出现所在的结点。

图 4-1(a)、(b)是经函数 compress() 处理前后的链表结构示例图。

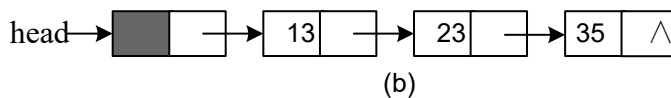
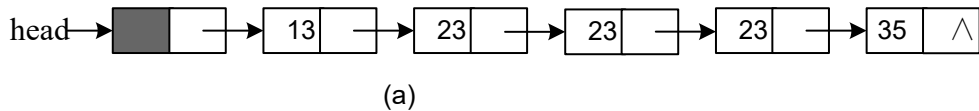


图 4-1

链表的结点类型定义如下：

```
typedef struct Node {  
    int data;  
    struct Node *next;  
}NODE;
```

[C 语言函数]

```
void compress(NODE *head)  
{  
    NODE *ptr,*q;  
    ptr = (1);    /* 取得第一个元素结点的指针 */  
    while ((2) && ptr -> next) {  
        q = ptr -> next;  
        while(q && (3)) { /* 处理重复元素 */  
            (4) = q -> next;  
            free(q);  
            q = ptr -> next;  
        }  
        (5) = ptr -> next;  
    }  
    /* end of while */  
} /* end of compress */
```

从下列 3 道试题（试题五至试题七）中任选 1 道解答。如果解答的试题数超过 1 道，则题号小的 1 道解答有效。

试题五（共 15 分）

阅读下列说明、图和 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]

已知四个类之间的关系如图 5-1 所示，分别对每个类的方法进行编号，例如 Shape 的 perimeter() 方法为 1 号，表示为“1:perimeter()”，Rectangle 类的 perimeter() 为 2 号，表示为“2:perimeter()”，依此类推，其中，每个类的 perimeter 方法都为虚函数且方法签名相同。

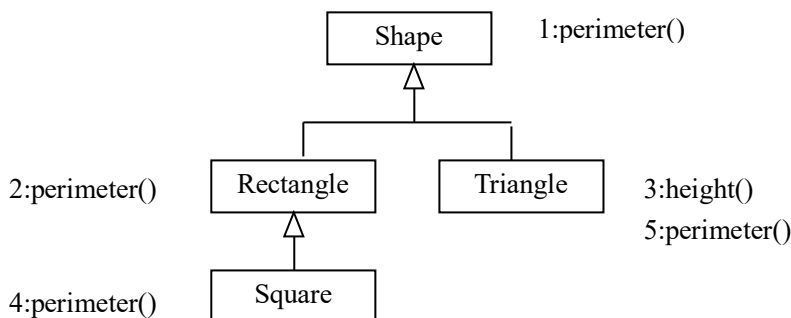


图 5-1 类图

[C++代码]

```
Triangle *tr = new Triangle();
Square *sq = new Square();
Shape *sh = tr;
```

[问题 1] 关于上述 C++代码中 sh 和 tr 的以下叙述中，哪两个是正确的（写出编号）。

- ① sh 和 tr 分别引用同一个对象；
- ② sh 和 tr 分别引用同一类型的不同的对象；
- ③ sh 和 tr 分别引用不同类型的不同对象；
- ④ sh 和 tr 分别引用同一个对象的不同拷贝；
- ⑤ sh 和 tr 所引用的内存空间是相同的。

[问题 2] 写出下面消息对应的方法编号（如果该消息错误或者没有对应的方法调用，请填写“无”）。

tr->height()	_____ (1)
sh->perimeter()	_____ (2)
sq->height()	_____ (3)
sq->perimeter()	_____ (4)
sh->height()	_____ (5)
tr->perimeter()	_____ (6)

[问题 3] 不考虑内存释放问题，下列赋值语句中哪两个是合法的（写出合法赋值语句的编号）。

- ① sq = sh; ② sh = tr; ③ tr = sq; ④ sq = tr; ⑤ sh = sq;

试题六（共 15 分）

阅读以下应用说明以及 Visual Basic 程序代码，将应填入__（n）__处的字句写在答题纸的对应栏内。

[应用说明]

某电视台拟开发应用程序来显示戏曲大赛中 1~4 号四位选手决赛的现场投票情况。该程序从现场观众中（不超过 2000 人）每 0.5 秒收集一次对这四位选手的支持票数，并在屏幕上动态显示这四位选手的票柱（以高度反映票数）与累计得票数，如图 6-1 所示。投票过程限时 30 秒，每名观众最多为 1 名选手投票。投票结束后系统通过比较各位选手的累计得票数，显示决赛结果：“*号胜出”（如有单个冠军）或“继续进行 PK”（如有多人获得相同的最高票数）。

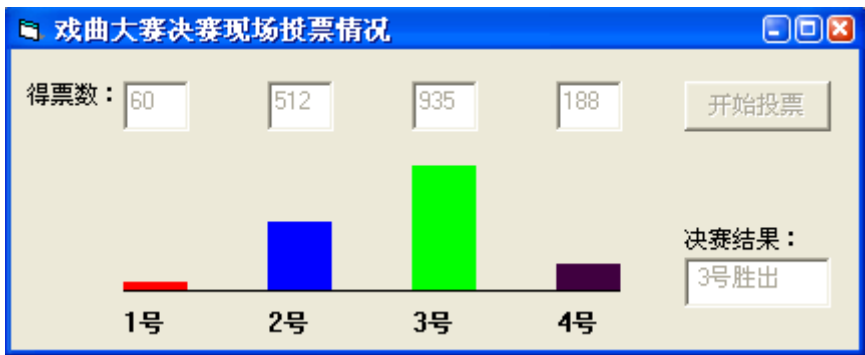


图 6-1

在开发该程序的过程中创建的主要控件及其初始属性值说明如下：

控件名	类型	用途	初始属性设置
ShpM (1 to 4)	形状数组	显示各选手得票情况	矩形，实心，高度 0
txtM(1 to 4)	文本框数组	显示各选手得票数	0
cmdStart	命令按钮	启动计票	标题：开始投票
txtResult	文本框	显示决赛结果	空
Tim1	计时器	每半秒收集处理一次	间隔时间 0.5 秒，关闭状态

该程序中设置公共变量 T 动态记录投票时间。四个形状 ShpM (1 to 4) 动态增长的底线固定。

[Visual Basic 程序代码]

```
Dim T As Integer      ' 声明全局变量
Private Sub Form_Load()
    For i = 1 To 4
        ShpM(i).Top = 2000 : ShpM(i).Height = 0      ' 初始票柱高度为 0
        TxtM(i).Text = 0
    Next i
    Tim1.Enabled = False : Tim1.Interval = 500 : T = 0
End Sub
```

```

Private Sub CmdStart_Click()
    Tim1.Enabled = True                ' 开始投票
    CmdStart.Enabled = False
End Sub

Private Sub Tim1_Timer()
    Dim n(1 To 4) As Integer            ' n(1 to 4) 为每次收集的票数
    Dim i As Integer, j As Integer
    Dim G As Integer                    ' G 用于计算最高票数
    Dim ng As Integer                   ' ng 用于计算冠军个数
    For i = 1 To 4
        n(i) = ...                      ' 收集 i 号选手的票数，此处省略
        TxtM(i).Text = TxtM(i).Text + n(i) ' 累计票（VB 能进行自动转换）
        ShpM(i).Top = ShpM(i).Top - n(i)
        ShpM(i).Height = ShpM(i).Height + (1) ' 算出票柱高度
    Next i
    T = T + 1                            ' 计时
    If T = 60 Then                        ' 投票时间到
        (2)                              ' 停止数据收集处理
        ng = 1
        G = TxtM(1).Text
        For i = 2 To 4
            If G < TxtM(i).Text Then
                G = TxtM(i).Text
                ng = (3)
                j = i
            Else
                If G = TxtM(i).Text Then ng = (4) ' 计算冠军个数
            End If
        Next i
        If ng = 1 Then
            txtResult.Text = (5) ' 单个冠军结果
        Else
            txtResult.Text = "继续进行 PK"
        End If
    End If
End Sub

```

试题七（共 15 分）

阅读下列说明、图和 Java 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]

已知四个类之间的关系如图 7-1 所示，分别对每个类的方法进行编号，例如 Shape 的 `perimeter()` 为 1 号，表示为“1:perimeter()”，Rectangle 类的 `perimeter()` 为 2 号，表示为“2:perimeter()”，依此类推，其中，每个类的 `perimeter` 方法签名相同。

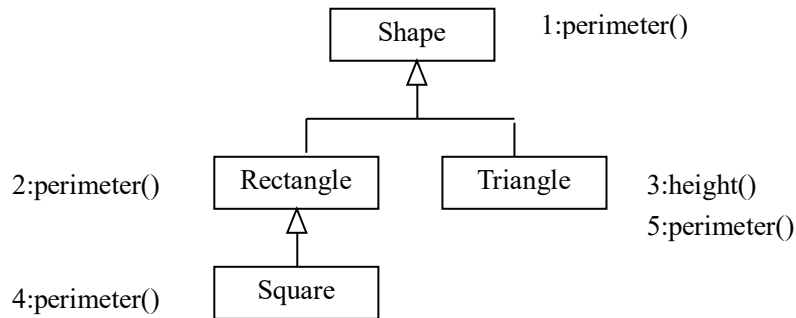


图 7-1 类图

[Java 代码]

```
Triangle tr = new Triangle();
Square sq = new Square();
Shape sh = tr;
```

[问题 1] 关于上述 Java 代码中 `sh` 和 `tr` 的以下叙述中，哪两个是正确的（写出编号）。

- ① `sh` 和 `tr` 分别引用同一个对象；
- ② `sh` 和 `tr` 分别引用同一类型的不同的对象；
- ③ `sh` 和 `tr` 分别引用不同类型的不同对象；
- ④ `sh` 和 `tr` 分别引用同一个对象的不同拷贝；
- ⑤ `sh` 和 `tr` 所引用的内存空间是相同的。

[问题 2] 写出下面消息对应的方法编号（如果该消息错误或者没有对应的方法调用，请填写“无”）。

<code>tr.height()</code>	<u> (1) </u>
<code>sh.perimeter()</code>	<u> (2) </u>
<code>sq.height()</code>	<u> (3) </u>
<code>sq.perimeter()</code>	<u> (4) </u>
<code>sh.height()</code>	<u> (5) </u>
<code>tr.perimeter()</code>	<u> (6) </u>

[问题 3] 下列赋值语句中哪两个是合法的（写出合法赋值语句的编号）。

- ① `sq = sh;` ② `sh = tr;` ③ `tr = sq;` ④ `sq = tr;` ⑤ `sh = sq;`