

全国计算机技术与软件专业技术资格（水平）考试

2009 年上半年 软件评测师 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

● 2009 年上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. 4 B. 5 C. 6 D. 7

(89) A. 20 B. 21 C. 22 D. 23

因为考试日期是“5 月 23 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）。

● 计算机的用途不同，对其部件的性能指标要求也有所不同。以科学计算为主的计算机，对(1)要求较高，而且应该重点考虑(2)。

- (1) A. 外存储器的读写速度 B. 主机的运算速度
C. I/O 设备的速度 D. 显示分辨率

- (2) A. CPU 的主频和字长，以及内存容量
B. 硬盘读写速度和字长
C. CPU 的主频和显示分辨率
D. 硬盘读写速度和显示分辨率

● (3)是指按内容访问的存储器。

- (3) A. 虚拟存储器 B. 相联存储器
C. 高速缓存 (Cache) D. 随机访问存储器

● 处理机主要由处理器、存储器和总线组成，总线包括(4)。

- (4) A. 数据总线、地址总线、控制总线 B. 并行总线、串行总线、逻辑总线
C. 单工总线、双工总线、外部总线 D. 逻辑总线、物理总线、内部总线

● 下面关于加密的说法中，错误的是(5)。

- (5) A. 数据加密的目的是保护数据的机密性
B. 加密过程是利用密钥和加密算法将明文转换成密文的过程
C. 选择密钥和加密算法的原则是保证密文不可能被破解
D. 加密技术通常分为非对称加密技术和对称密钥加密技术

● 下面关于防火墙功能的说法中，不正确的是(6)。

- (6) A. 防火墙能有效防范病毒的入侵
B. 防火墙能控制对特殊站点的访问
C. 防火墙能对进出的数据包进行过滤
D. 防火墙能对部分网络攻击行为进行检测和报警

● 下面关于漏洞扫描系统的叙述，错误的是(7)。

- (7) A. 漏洞扫描系统是一种自动检测目标主机安全弱点的程序
B. 黑客利用漏洞扫描系统可以发现目标主机的安全漏洞
C. 漏洞扫描系统可以用于发现网络入侵者
D. 漏洞扫描系统的实现依赖于系统漏洞库的完善

● 软件工程每一个阶段结束前，应该着重对可维护性进行复审。在系统设计阶段的复审期间，应该从(8)出发，评价软件的结构和过程。

- (8) A. 指出可移植性问题以及可能影响软件维护的系统界面

- B. 容易修改、模块化和功能独立的目的
- C. 强调编码风格和内部说明文档
- D. 可测试性

● 计算机感染特洛伊木马后的典型现象是__ (9) __。

- (9) A. 程序异常退出 B. 有未知程序试图建立网络连接
C. 邮箱被垃圾邮件填满 D. Windows系统黑屏

● 关于软件著作权产生的时间，下面表述正确的是__ (10) __。

- (10) A. 自作品首次公开发表时
B. 自作者有创作意图时
C. 自作品得到国家著作权行政管理部门认可时
D. 自作品完成创作之日

● 程序员甲与同事乙在乙家探讨甲近期编写的程序，甲表示对该程序极不满意，说要弃之重写，并将程序手稿扔到乙家垃圾筒。后来乙将甲这一程序稍加修改，并署名发表。以下说法正确的是__ (11) __。

- (11) A. 乙的行为侵犯了甲的软件著作权
B. 乙的行为没有侵犯甲的软件著作权，因为甲已将程序手稿丢弃
C. 乙的行为没有侵犯甲的著作权，因为乙已将程序修改
D. 甲没有发表该程序并弃之，而乙将程序修改后发表，故乙应享有著作权

● 零件关系P(零件名，条形码，供应商，产地，价格)中的__ (12) __属性可以作为该关系的主键。查询产于西安且名称为“P2”的零件，结果以零件名、供应商及零件价格分列表示，对应的SQL语句为：

```
SELECT 零件名, 供应商, 价格  
FROM P  
WHERE 零件名='P2' AND __ (13) __;
```

- (12) A. 零件名 B. 条形码 C. 产地 D. 供应商
(13) A. 条形码=西安 B. 条形码='西安'
C. 产地=西安 D. 产地='西安'

● 软件风险一般包含__ (14) __两个特性。

- (14) A. 救火和危机管理 B. 已知风险和未知风险
C. 不确定性和损失 D. 员工和预算

● 在采用面向对象技术构建软件系统时，很多敏捷方法都建议的一种重要的设计活动是(15)，它是一种重新组织的技术，可以简化构件的设计而无需改变其功能或行为。

- (15) A. 精化 B. 设计类 C. 重构 D. 抽象

● 一个软件开发过程描述了“谁做”、“做什么”、“怎么做”和“什么时候做”，RUP用(16)来表述“谁做”。

- (16) A. 角色 B. 活动 C. 制品 D. workflow

● 瀑布模型表达了一种系统的、顺序的软件开发方法。以下关于瀑布模型的叙述中，正确的是(17)。

- (17) A. 瀑布模型能够非常快速地开发大规模软件项目
B. 只有很大的开发团队才使用瀑布模型
C. 瀑布模型已不再适合于现今的软件开发环境
D. 瀑布模型适用于软件需求确定，开发过程能够采用线性方式完成的项目

● 一个软件系统的生存周期包含可行性分析和项目开发计划、需求分析、设计（概要设计和详细设计）、编码、测试和维护等活动，其中(18)是软件工程的技术核心，其任务是确定如何实现软件系统。

- (18) A. 可行性分析和项目开发计划 B. 需求分析
C. 设计 D. 编码

● 程序中常采用变量表示数据，变量具有名、地址、值、作用域、生存期等属性。关于变量的叙述，(19)是错误的。

- (19) A. 根据作用域规则，在函数中定义的变量只能在函数中引用
B. 在函数中定义的变量，其生存期为整个程序执行期间
C. 在函数中定义的变量不能与其所在函数的形参同名
D. 在函数中定义的变量，其存储单元在内存的栈区

● 函数调用时，基本的参数传递方式有传值与传地址两种，(20)。

- (20) A. 在传值方式下，形参将值传给实参
B. 在传值方式下，实参不能是数组元素
C. 在传地址方式下，形参和实参间可以实现数据的双向传递
D. 在传地址方式下，实参可以是任意的变量和表达式

● 已知某高级语言源程序 A 经编译后得到机器 C 上的目标程序 B，则(21)。

- (21) A. 对 B 进行反编译，一般不能还原出源程序 A
B. 对 B 进行反汇编，不能得到与源程序 A 等价的汇编程序代码

- C. 对 B 进行反编译，得到的是源程序 A 的变量声明和算法流程
- D. 对 A 和 B 进行交叉编译，可以产生在机器 C 上运行的动态链接库

● 下面关于程序语言的叙述，错误的是 (22)。

- (22) A. 脚本语言属于动态语言，其程序结构可以在运行中改变
- B. 脚本语言一般通过脚本引擎解释执行，不产生独立保存的目标程序
- C. php、JavaScript 属于静态语言，其所有成分可在编译时确定
- D. C 语言属于静态语言，其所有成分可在编译时确定

● 在 Windows XP 操作系统中，用户利用“磁盘管理”程序可以对磁盘进行初始化、创建卷，(23)。通常将“C:\Windows\myprogram.exe”文件设置成只读和隐藏属性，以便控制用户对该文件的访问，这一级安全管理称之为 (24) 安全管理。

- (23) A. 但只能使用 FAT 文件系统格式化卷
- B. 但只能使用 FAT 32 文件系统格式化卷
- C. 但只能使用 NTFS 文件系统格式化卷
- D. 可以选择使用 FAT32 或 NTFS 文件系统格式化卷

- (24) A. 文件级 B. 目录级 C. 用户级 D. 系统级

● (25) 属于系统软件，它直接执行高级语言源程序或与源程序等价的某种中间代码。

- (25) A. 编译程序 B. 预处理程序 C. 汇编程序 D. 解释程序

● 设系统中有 R 类资源 m 个，现有 n 个进程互斥使用。若每个进程对 R 资源的最大需求为 w，那么当 m、n、w 取下表的值时，对于下表中的 a~e 五种情况，(26) 两种情况可能会发生死锁。对于这两种情况，若将 (27)，则不会发生死锁。

	a	B	c	d	e
m	2	2	2	4	4
n	1	2	2	3	3
w	2	1	2	2	3

- (26) A. a 和 b B. b 和 c C. c 和 d D. c 和 e

- (27) A. n 加 1 或 w 加 1 B. m 加 1 或 w 减 1
- C. m 减 1 或 w 加 1 D. m 减 1 或 w 减 1

● 在软件开发过程中，常采用图形表示相关的信息，(28) 不用于表示软件模块的执行过程。

- (28) A. N-S 盒图 B. E-R 图 C. PAD 图 D. 程序流程图

● 软件能力成熟度模型 (CMM) 将软件能力成熟度自低到高依次划分为 5 级。目前，

达到 CMM 第 3 级（已定义级）是许多组织努力的目标，该级的核心是（29）。

- (29) A. 建立基本的项目管理和实践来跟踪项目费用、进度和功能特性
- B. 使用标准开发过程（或方法论）构建（或集成）系统
- C. 管理层寻求更主动地应对系统的开发问题
- D. 连续地监督和改进标准化的系统开发过程

● RUP 在每个阶段都有主要目标，并在结束时产生一些制品。在（30）结束时产生“在适当的平台上集成的软件产品”。

- (30) A. 初期阶段 B. 精化阶段 C. 构建阶段 D. 移交阶段

● 关于软件测试，（31）的叙述是正确的。

- ① 测试开始越早，越有利于发现软件缺陷
- ② 采用正确的测试用例设计方法，软件测试可以做到穷举测试
- ③ 测试覆盖度和测试用例数量成正比
- ④ 软件测试的时间越长越好

- (31) A. ④ B. ① C. ②、③ D. ①、③

● 系统功能测试过程中，验证需求可以正确实现的测试用例称为（32）。

- (32) A. 业务流程测试用例 B. 功能点测试用例
- C. 通过测试用例 D. 失败测试用例

● （33）不属于功能测试用例构成元素。

- (33) A. 测试数据 B. 测试步骤 C. 预期结果 D. 实测结果

● 针对电子政务类应用系统的功能测试，为设计有效的测试用例，应（34）。

- (34) A. 使业务需求的覆盖率达到 100%
- B. 利用等价类法模拟核心业务流程的正确执行
- C. 对一个业务流程的测试用例设计一条验证数据
- D. 经常使用边界值法验证界面输入值

● （35）测试用例设计方法既可以用于黑盒测试，也可以用于白盒测试。

- (35) A. 边界值法 B. 基本路径法 C. 正交试验设计法 D. 逻辑覆盖法

● 对“功能测试的回归测试经常要多次重复”的正确理解是（36）。

- (36) A. 回归测试应该执行初测时所用的全部测试用例
- B. 回归测试只要执行发现缺陷的那些测试用例即可
- C. 通过多次的回归测试可以发现所有缺陷
- D. 回归测试就是验收测试

● 功能测试执行过后一般可以确认系统的功能缺陷，缺陷的类型包括 (37)。

- ① 功能不满足隐性需求
- ② 功能实现不正确
- ④ 功能易用性不好
- ③ 功能不符合相关的法律法规

(37) A. ① B. ①②③ C. ②③④ D. ②

● 以下关于软件测试的概念，正确的是 (38)。

- (38) A. 软件测试的目的是想证实在一个给定的外部环境中软件的逻辑正确性，即保证软件以正确的方式来做这个事件
- B. 软件质量保证的基本措施就是对软件进行确认测试
 - C. 软件测试的对象不仅仅是程序，文档、数据和规程都是软件测试的对象
 - D. 单元测试可检验程序单元或部件的接口关系，应能发现并排除在模块连接中可能发生的问题

● 以下不正确的软件测试原则是 (39) 。

- (39) A. 软件测试可以发现软件潜在的缺陷
- B. 所有的软件测试都可追溯到用户需求
 - C. 测试应尽早不断地执行
 - D. 程序员应避免测试自己的程序

● 在编码阶段对系统执行的测试类型主要包括单元测试和集成测试，(40) 属于单元测试的内容。

- (40) A. 接口数据测试 B. 局部数据测试
- C. 模块间时序测试 D. 全局数据测试

● 以下关于软件测试概念的叙述，不正确的是 (41)。

- (41) A. 软件失效指软件运行时产生了一种不希望或不可接受的内部行为
- B. 软件功能实现超出了产品说明书的规定说明软件存在缺陷
 - C. 测试目的是为了发现软件缺陷与错误，也是对软件质量进行度量和评估
 - D. 在软件生命周期各个阶段都可能产生错误

● 以下关于软件测试分类定义的叙述，不正确的是 (42)。

- (42) A. 软件测试可分为单元测试、集成测试、确认测试、系统测试、验收测试
- B. 确认测试是在模块测试完成的基础上，将所有的程序模块进行组合并验证其是否满足用户需求的过程
 - C. 软件测试可分为白盒测试和黑盒测试
 - D. 系统测试是将被测软件作为整个基于计算机系统的一个元素，与计算机硬件、外设、某些支持软件、数据和人员等其他系统元素结合在一起进行测试的过程

● 正确的集成测试描述包括 (43)。

①集成测试也叫做组装测试，通常是在单元测试的基础上，将模块按照设计说明书要求进行组装和测试的过程。

②自顶向下的增殖方式是集成测试的一种组装方式，它能较早地验证主要的控制和判断点，对于输入输出模块、复杂算法模块中存在的错误能够较早地发现。

③集成测试的目的在于检查被测模块能否正确实现详细设计说明中的模块功能、性能、接口和设计约束等要求

④集成测试需要重点关注各个模块之间的相互影响，发现并排除全局数据结构问题

(43) A. ①② B. ②③ C. ①④ D. ②④

● 以下关于软件质量特性测试的叙述，正确的是 (44)。

①成熟性测试是检验软件系统故障，或违反指定接口的情况下维持规定的性能水平有关的测试工作

②功能性测试是检验适合性、准确性、互操作性、安全保密性、功能依从性的测试工作

③易学性测试是检查系统中用户为操作和运行控制所花努力有关的测试工作

④效率测试是指在规定的条件下产品执行其功能时，对时间消耗及资源利用的测试工作

(44) A. ①②③④ B. ①④ C. ①③④ D. ②④

● 对软件可靠性的理解，正确的是 (45)。

①软件可靠性是指在指定条件下使用时，软件产品维持规定的性能级别的能力

②软件可靠性的种种局限是由于随着时间的推移，软件需求和使用方式发生了变化

③软件可靠性包括成熟性、有效性、容错性、易恢复性等质量特性

④针对软件可靠性中的容错性特性应测试软件失效防护能力

(45) A. ①③ B. ②③ C. ①④ D. ①②③④

● 软件可移植性应从如下 (46) 方面进行测试。

(46) A. 适应性、易安装性、共存性、易替换性

B. 适应性、易安装性、可伸缩性、易替换性

C. 适应性、易安装性、兼容性、易替换性

D. 适应性、成熟性、兼容性、易替换性

● 以下关于基于 V&V 原理的 W 模型的叙述中，(47) 是错误的。

(47) A. W 模型指出当需求被提交后，就需要确定高级别的测试用例来测试这些需求，当详细设计编写完成后，即可执行单元测试

B. 根据 W 模型要求，一旦有文档提供，就要及时确定测试条件、编写测试

用例

- C. 软件测试贯穿于软件定义和开发的整个期间
- D. 程序、需求规格说明、设计规格说明都是软件测试的对象

● 以下说法不正确的选项包括 (48)。

- ①软件测试不仅仅指测试的执行，还包括很多其他的活动
- ②软件测试是一个独立的流程，贯穿产品整个生命周期，与其他流程并发地进行
- ③应用 H 模型有利于资源调配，有助于跟踪测试投入的流向
- ④H 模型指出，单元测试、集成测试、系统测试不存在严格的次序关系，各层次之间的测试存在反复触发、迭代和增量关系等

(48) A. ①③ B. ②③ C. ①④ D. 无

● 以下软件质量保证的目标中，(49) 是错误的。

- (49) A. 通过监控软件开发过程来保证产品质量
- B. 保证开发出来的软件和软件开发过程符合相应标准与规程，不存在软件缺陷
- C. 保证软件产品、软件过程中存在的问题得到处理，必要时将问题反映给高级管理者
- D. 确保项目组制定的计划、标准和规程适合项目组需要，同时满足评审和审计需要

● (50) 不会影响测试质量。

- (50) A. 用户需求频繁变化 B. 测试流程不规范
- C. 采用背靠背测试方式 D. 测试周期被压缩

● (51) 不属于测试人员编写的文档。

- (51) A. 缺陷报告 B. 测试环境配置文档
- C. 缺陷修复报告 D. 测试用例说明文档

● GB/T 16260-2006《软件工程 产品质量》规定的软件产品使用质量特性包括：(52)。

- (52) A. 适应性、生产率、可靠性、满意度
- B. 有效性、生产率、安全性、满意度
- C. 有效性、可靠性、适应性、满意度
- D. 适应性、适用性、效率、满意度

● GB 17859-1999《计算机信息系统安全保护等级划分准则》中将计算机安全保护

划分为 (53) 个级别。

- (53) A. 3 B. 4 C. 5 D. 6

● 假设在程序控制流图中，有 12 条边，8 个节点，则确保程序中每个可执行语句至少执行一次所必需的测试用例数目的上限是 (54)。

- (54) A. 12 B. 8 C. 6 D. 4

● 关于白盒测试的测试用例设计方法叙述，正确的是 (55)。

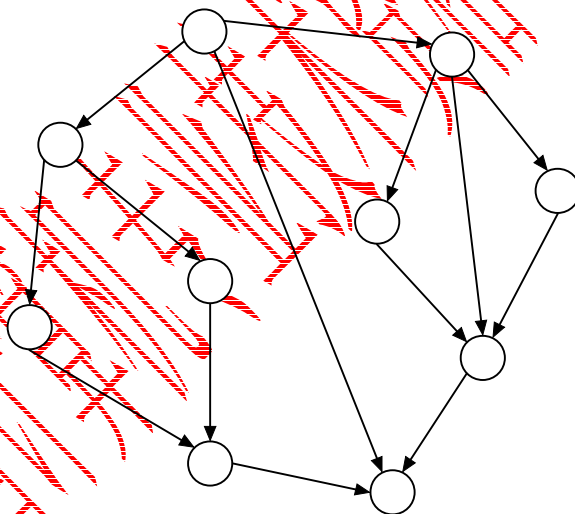
(55) A. 完成 SC (语句判定) 所需的测试用例数目一定多于完成 DC (逻辑判定) 所需的测试用例数目

B. 达到 100%CC (条件判定) 要求就一定能够满足 100%SC 的要求

C. 达到 100%CDC (条件判定组合覆盖) 要求就一定能够满足 100%CC 的要求

D. 任何情况下，都可以达到 100%路径覆盖的要求

● 以下控制流图的圈复杂度 $V(g)$ 为 (56)。



- (56) A. 4 B. 6 C. 8 D. 10

● 针对程序段：IF (A || B || C) THEN W=W/X，对于(A, B, C)的取值，(57) 测试用例能够满足 MCDC (修正条件逻辑判定) 的要求。

- (57) A. (F, T, T) (T, F, T) (T, F, F) (T, T, F)
B. (T, F, F) (T, T, F) (F, T, T) (F, F, F)
C. (T, F, F) (T, T, F) (F, T, T) (F, F, T)
D. (T, F, F) (F, T, F) (F, F, T) (F, F, F)

● 针对下列程序段，需要(58)个测试用例可以满足分支覆盖的要求。

```
int IsLeap(int year)
{
    if ( year % 4 == 0 )
    {
        if ( ( year % 100 == 0 )
            {
                if ( year % 400 == 0 )
                    leap = 1;
                else
                    leap = 0;
            }
        else
            leap = 1;
    }
    else
        leap = 0;
    return leap;
}
```

(58) A. 3 B. 4 C. 6 D. 7

● 黑盒测试中，(59)是根据输出对输入的依赖关系设计测试用例。

(59) A. 基本路径法 B. 等价类 C. 因果图 D. 功能图法

● Web 应用系统负载压力测试中，(60)不是衡量业务执行效率的指标。

(60) A. 并发请求数 B. 每秒点击率
 C. 交易执行吞吐量 D. 交易执行响应时间

● 软件测试的基本方法包括白盒测试和黑盒测试方法，以下关于二者之间关联的叙述，错误的是(61)。

(61) A. 黑盒测试与白盒测试是设计测试用例的两种基本方法
 B. 在集成测试阶段是采用黑盒测试与白盒测试相结合的方法
 C. 针对相同的系统模块，执行黑盒测试和白盒测试对代码的覆盖率都能够达到 100%
 D. 应用系统负载压力测试一般采用黑盒测试方法

● 为验证某音乐会订票系统是否能够承受大量用户同时访问，测试工程师一般采用(62)测试工具。

- (62) A. 故障诊断 B. 代码 C. 负载压力 D. 网络仿真

● (63) 不属于网站渗透测试的内容。

- (63) A. 防火墙日志审查 B. 防火墙远程探测与攻击
C. 跨站攻击 D. SQL 注入

● 能够主动采集信息，分析网络攻击行为和误操作的实时保护策略是指(64)。

- (64) A. 安全日志 B. 入侵检测 C. 隔离防护 D. 防火墙

● 下列设备和技术中，(65) 不属于数据安全策略范畴。

- (65) A. SAN B. 异地容灾 C. 数字证书 D. 双机容错

● 以下对用户口令进行限定的措施中，(66) 对提高安全性来说是无用的。

- (66) A. 最小改变口令的时限
B. 最短口令长度
C. 口令不能全为数字或小写字母之外的字符
D. 首次登录必需改变口令

● (67) 不属于用户认证机制。

- (67) A. 用户名和密码 B. 数字证书 C. SSL D. User Key

● 在 Windows 系统中设置默认路由的作用是(68)。

- (68) A. 当主机接收到一个访问请求时首先选择的路由
B. 当没有其它路由可选时最后选择的路由
C. 访问本地主机的路由
D. 必须选择的路由

● 集线器是一种物理层联网设备，下面有关集线器的叙述中，错误的是(69)。

- (69) A. 集线器没有自动寻址能力
B. 连接在集线器上的故障设备不会影响其他结点通信
C. 集线器可以隔离广播风暴
D. 连接在集线器上的所有结点构成一个冲突域

● 用户访问某 Web 网站，浏览器上显示“HTTP-404”错误，则故障原因是(70)。

- (70) A. 默认路由器配置不当 B. 所请求当前页面不存在
C. Web 服务器内部出错 D. 用户无权访问

● For nearly ten years, the Unified Modeling Language (UML) has been the industry standard for visualizing, specifying, constructing, and documenting the (71) of a software-intensive system. As the (72) standard modeling language, the UML facilitates communication and reduces confusion among project (73). The recent standardization of UML 2.0 has further extended the language's scope and viability. Its inherent expressiveness allows users to (74) everything from enterprise information systems and distributed Web-based applications to real-time embedded systems.

The UML is not limited to modeling software. In fact, it is expressive enough to model (75) systems, such as workflow in the legal system, the structure and behavior of a patient healthcare system, software engineering in aircraft combat systems, and the design of hardware.

To understand the UML, you need to form a conceptual model of the language, and this requires learning three major elements: the UML's basic building blocks, the rules that dictate how those building blocks may be put together, and some common mechanisms that apply throughout the UML.

- | | | | |
|----------------------|-----------------|--------------|-----------------|
| (71) A. classes | B. components | C. sequences | D. artifacts |
| (72) A. real | B. legal | C. de facto | D. illegal |
| (73) A. investors | B. developers | C. designers | D. stakeholders |
| (74) A. model | B. code | C. test | D. modify |
| (75) A. non-hardware | B. non-software | C. hardware | D. software |