

# 全国计算机技术与软件专业技术资格（水平）考试

## 2007 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

● 2007 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. 23 | B. 24 | C. 25 | D. 26 |

因为考试日期是“5 月 26 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）。

● (1) 不属于计算机控制器中的部件。

- (1) A. 指令寄存器 IR B. 程序计数器 PC  
C. 算术逻辑单元 ALU D. 程序状态字寄存器 PSW

● 在 CPU 与主存之间设置高速缓冲存储器 Cache 的目的是为了 (2)。

- (2) A. 扩大主存的存储容量 B. 提高 CPU 对主存的访问效率  
C. 既扩大主存容量又提高存取速度 D. 提高外存储器的速度

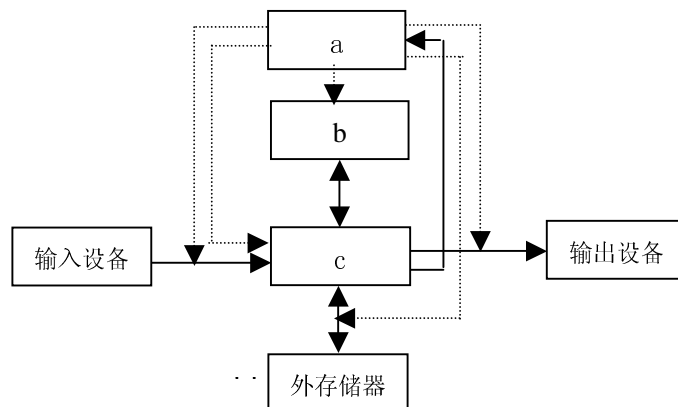
● 下面的描述中, (3) 不是 RISC 设计应遵循的设计原则。

- (3) A. 指令条数应少一些  
B. 寻址方式尽可能少  
C. 采用变长指令, 功能复杂的指令长度长而简单指令长度短  
D. 设计尽可能多的通用寄存器

● 系统响应时间和作业吞吐量是衡量计算机系统性能的重要指标。对于一个持续处理业务的系统而言, 其 (4)。

- (4) A. 响应时间越短, 作业吞吐量越小 B. 响应时间越短, 作业吞吐量越大  
C. 响应时间越长, 作业吞吐量越大 D. 响应时间不会影响作业吞吐量

● 计算机各功能部件之间的合作关系如下图所示。假设图中虚线表示控制流, 实线表示数据流, 那么 a、b 和 c 分别表示 (5)。



- (5) A. 控制器、内存储器 and 运算器 B. 控制器、运算器和内存储器  
C. 内存储器、运算器和控制器 D. 内存储器、控制器和运算器

● 下面关于编程语言的各种说法中，(6)是正确的。

- (6) A. 由于C语言程序是由函数构成的，所以也属于函数型语言  
B. Smalltalk、C++、Java、C#都是面向对象语言  
C. 函数型语言适用于编写高速计算的程序，常用于超级计算机的模拟计算  
D. 逻辑型语言是在Client/Server系统中用于实现负载分散的程序语言

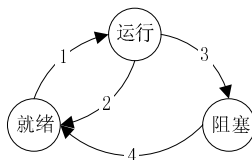
● 在面向对象的语言中，(7)。

- (7) A. 类的实例化是指对类的实例分配存储空间  
B. 每个类都必须创建一个实例  
C. 每个类只能创建一个实例  
D. 类的实例化是指对类进行初始化

● 在统一建模语言(UML)中，描述系统与外部系统及用户之间交互的图是(8)。

- (8) A. 类图                      B. 用例图                      C. 对象图                      D. 协作图

● 某系统的进程状态转换如下图所示。图中1、2、3和4分别表示引起状态转换时的不同原因。原因4是由于(9)；一个进程状态转换会引起另一个进程状态转换的是(10)。



- (9) A. 就绪进程被调度                      B. 运行进程执行了P操作  
C. 阻塞进程等待的事件发生了              D. 运行进程时间片到了  
(10) A. 1→2                      B. 2→1                      C. 3→2                      D. 2→4

● 在操作系统中，虚拟设备通常采用(11)设备来模拟低速设备（如打印机）。

- (11) A. Spooling技术，利用磁带              B. Spooling技术，利用磁盘  
C. 脱机批处理技术，利用磁盘              D. 通道技术，利用磁带

● 某文件管理系统在磁盘上建立了位示图(bitmap)，记录磁盘的使用情况。若系统中字长为32位，磁盘上的物理块依次编号为：0、1、2、…，那么8192号物理块的使用情况在位示图中的第(12)个字中有所描述。

- (12) A. 256                      B. 257                      C. 512                      D. 1024

● 在数据库管理系统中，(13)不属于安全性控制机制。

- (13) A. 完整性约束              B. 视图                      C. 密码验证                      D. 用户授权

● 某供应商数据库中的供应关系为 SPJ (供应商号, 零件号, 工程号, 数量), 如下命令查询某工程至少用了 3 家供应商 (包含 3 家) 供应的零件的平均数量, 并按工程号的降序排列。

```
SELECT 工程号, (14) FROM SPJ
GROUP BY 工程号
(15)
```

ORDER BY 工程号 DESC;

- (14) A. AVG (数量) At 平均数量                      B. AVG (数量) AS 平均数量  
C. 平均数量 At AVG (数量)                      D. 平均数量 AS AVG (数量)
- (15) A. HAVING COUNT (DISTINCT (供应商号)) >2  
B. Where COUNT (供应商号) >2  
C. HAVING (DISTINCT (供应商号)) >2  
D. Where 供应商号 >2

● 设关系模式 R (A, B, C), 传递依赖指的是 (16); 下列结论错误的是 (17) 。

- (16) A. 若  $A \rightarrow B, B \rightarrow C$ , 则  $A \rightarrow C$                       B. 若  $A \rightarrow B, A \rightarrow C$ , 则  $A \rightarrow BC$   
C. 若  $A \rightarrow C$ , 则  $AB \rightarrow C$                       D. 若  $A \rightarrow BC$ , 则  $A \rightarrow B, A \rightarrow C$
- (17) A. 若  $A \rightarrow BC$ , 则  $A \rightarrow B, A \rightarrow C$                       B. 若  $A \rightarrow B, A \rightarrow C$ , 则  $A \rightarrow BC$   
C. 若  $A \rightarrow C$ , 则  $AB \rightarrow C$                       D. 若  $AB \rightarrow C$ , 则  $A \rightarrow C, B \rightarrow C$

● 软件开发人员可以用 (18) 软件编写和修改程序。

- (18) A. 预处理              B. 文本编辑              C. 链接              D. 编译

● 统一过程 (UP) 是一种用例驱动的迭代式增量开发过程, 每次迭代过程中主要的工作流包括捕获需求、分析、设计、实现和测试等。这种软件过程用例图 (Use Case Diagram) 是通过 (19) 得到的。

- (19) A. 捕获需求              B. 分析              C. 设计              D. 实现

● 关于原型化开发方法的叙述中, 不正确的是 (20)。

- (20) A. 原型化方法适应于需求不明确的软件开发  
B. 在开发过程中, 可以废弃不用早期构造的软件原型  
C. 原型化方法可以直接开发出最终产品  
D. 原型化方法利于确认各项系统服务的可用性

● CMM 模型将软件过程的成熟度分为 5 个等级。在 (21) 使用定量分析来不断地改进和管理软件过程。

- (21) A. 优化级              B. 管理级              C. 定义级              D. 可重复级

● 软件 (22) 的提高, 有利于软件可靠性的提高。

(22) A. 存储效率 B. 执行效率 C. 容错性 D. 可移植性

● 序言性注释是指在每个程序或模块开头的一段说明, 起辅助理解程序的作用, 一般包括: 程序的表示、名称和版本号; 程序功能描述; 接口与界面描述; 输入/输出数据说明; 开发历史; 与运行环境有关的信息等。下列叙述中不属于序言性注释的是 (23)。

(23) A. 程序对硬件、软件资源的要求  
B. 重要变量和参数说明  
C. 嵌入在程序中的 SQL 语句  
D. 程序开发的原作者、审查者、修改者、编程日期等

● LOC 是软件规模的一种量度, 它表示 (24)。

(24) A. 软件功能数 B. 源代码行数 C. 每单位成本数 D. 工作量

● 面向对象分析与设计是面向对象软件开发过程中的两个重要阶段, 下列活动中, (25) 不属于面向对象分析阶段。

(25) A. 构建分析模型 B. 识别分析类 C. 确定接口规格 D. 评估分析模型

● 面向对象分析需要找出软件需求中客观存在的所有实体对象(概念), 然后归纳、抽象出实体类。(26) 是寻找实体对象的有效方法之一。

(26) A. 会议调查 B. 问卷调查 C. 电话调查 D. 名词分析

● 在“模型—视图—控制器”(MVC) 模式中, (27) 主要表现用户界面, (28) 用来描述核心业务逻辑。

(27) A. 视图 B. 模型 C. 控制器 D. 视图和控制器  
(28) A. 视图 B. 模型 C. 控制器 D. 视图和控制器

● 在进行面向对象设计时, 采用设计模式能够 (29)。

(29) A. 复用相似问题的相同解决方案 B. 改善代码的平台可移植性  
C. 改善代码的可理解性 D. 增强软件的易安装性

● 下面给出了四种设计模式的作用:

外观 (Façade): 为子系统中的一组功能调用提供一个一致的接口, 这个接口使得这一子系统更加容易使用;

装饰 (Decorate): 当不能采用生成子类的方法进行扩充时, 动态地给一个对象添加一些额外的功能;

单件 (Singleton): 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;

模板方法 (Template Method): 在方法中定义算法的框架, 而将算法中的一些操作步骤延迟到子类中实现。

请根据下面叙述的场景选用适当的设计模式。若某面向对象系统中的某些类有且只有一个实例, 那么采用 (30) 设计模式能够有效达到该目的; 该系统中的某子模块需要为其他模块提供访问不同数据库系统 (Oracle、SQL Server、DB2 UDB 等) 的功能, 这些数据库系统提供的访问接口有一定的差异, 但访问过程却都是相同的, 例如, 先连接数据库、再打开数据库、最后对数据进行查询, (31) 设计模式可抽象出相同的数据库访问过程; 系统中的文本显示类 (TextView) 和图片显示类 (PictureView) 都继承了组件类 (Component), 分别显示文本和图片内容, 现需要构造带有滚动条、或者带有黑色边框、或者既有滚动条又有黑色边框的文本显示控件和图片显示控件, 但希望最多只增加三个类, (32) 设计模式可以实现该目的。

- (30) A. 外观            B. 装饰            C. 单件            D. 模板方法  
(31) A. 外观            B. 装饰            C. 单件            D. 模板方法  
(32) A. 外观            B. 装饰            C. 单件            D. 模板方法

● 函数  $t()$ 、 $f()$  的定义如下所示。若调用函数  $t()$  时传递给  $x$  的值为 3, 并且调用函数  $f()$  时, 第一个参数采用传值 (call by value) 方式, 第二个参数采用传引用 (call by reference) 方式, 则函数  $t()$  的返回值为 (33)。

$t(\text{int } x)$

```
int a;  
a = 3*x - 1;  
f(x, a);  
return a+x;
```

$f(\text{int } r, \text{int } s)$

```
int x;  
x = 2*r + 1 ; s = x*r;  
r = s - x;  
return;
```

- (33) A. 35            B. 24            C. 22            D. 11

● 程序设计语言中 (34)。

- (34) A. while 循环语句的执行效率比 do-while 循环语句的执行效率高  
B. while 循环语句的循环体执行次数比循环条件的判断次数多 1, 而 do-while 语句的循环体执行次数比循环条件的判断次数少 1  
C. while 语句的循环体执行次数比循环条件的判断次数少 1, 而 do-while 语句的循环体执行次数比循环条件的判断次数多 1  
D. while 语句的循环体执行次数比循环条件的判断次数少 1, 而 do-while 语句的循环体执行次数等于循环条件的判断次数

● 两名以上的申请人分别就同样的发明创造申请专利的, 专利权授权 (35)。

- (35) A. 最先发明的人            B. 最先申请的人  
C. 所有申请的人            D. 协商后的申请人

● 下列叙述中 (36) 是正确的。

(36) A. 压迫测试: 提供条件任软件发挥, 最大限度地发掘软件的能力

B. 重复测试：使软件在不够理想的条件下运行，观察软件对外部资源的要求和依赖的程度

C. 重复测试：不断执行同样的操作，这种反复测试的主要原因是看内存是否不足

D. 完整 C/S 体系结构测试, 只包括网络运行和性能测试

● 根据 ANSI / IEEE829 标准, 以下(37)属于《测试程序说明》中程序步骤的内容。

### ① 启动

## ② 目的

### ③ 日志

#### ④ 设置

(37) A. ①、②、③

B. ①、③、④

C. ②、③

D. ①、②、③、④

● 以下关于功能测试用例的意义的叙述，正确的是（38）。

### ① 避免盲目测试并提高测试效率

② 令软件测试的实施重点突出、目的明确

③ 在回归测试中无需修正测试用例便可继续开展测试工作

#### ④ 测试用例的通用化和复用化使软件测试易于开展

(38) A. ①、②、③

B. ①、③

C. ②、③

D. ①、②、④

● 用等价类法划分 Windows 文件名称，应该分成 (39) 个等价区间。

(39) A. 2

B. 3

C. 4

D. 6

● 用边界值分析法，假定 X 为整数， $10 \leq X \leq 100$ ，那么 X 在测试中应该取(40)边界值。

(40) A.  $X=10$ ,  $X=100$

B.  $X=9$ ,  $X=10$ ,  $X=100$ ,  $X=101$

C. X=10, X=11, X=99, X=100

D.  $X=9$ ,  $X=10$ ,  $X=50$ ,  $X=100$

● (41) 不是易用性测试包括的内容。

(41) A. 安装测试

## B. 界面测试

### C. 菜单测试

#### D. 文档测试

● (42) 不是文档测试包括的内容。

(42) A. 合同文档

## B. 开发文档

### C. 管理文档

#### D. 用户文档

● 针对用户手册的测试，(43) 描述不正确。

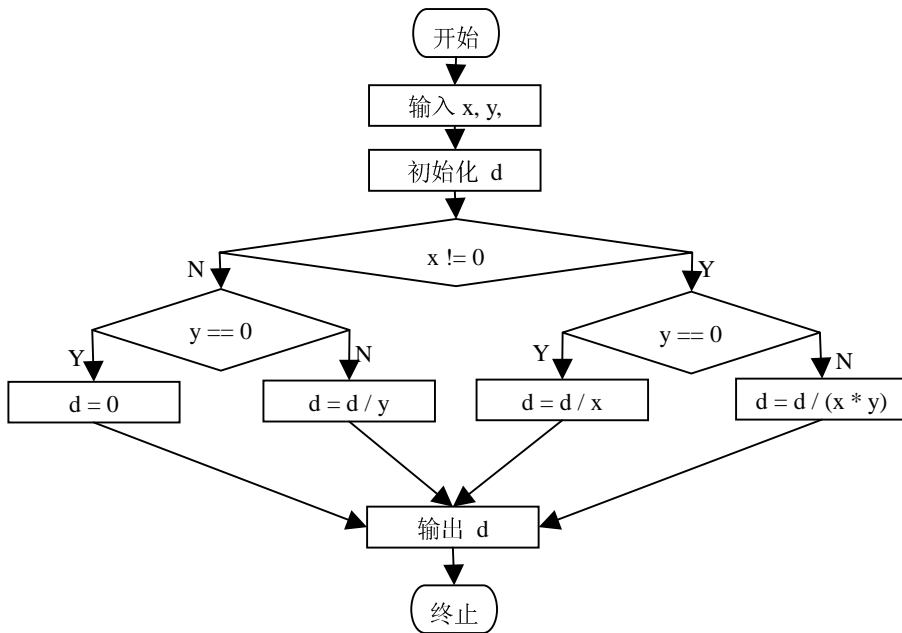
(43) A. 准确地按照手册的描述使用程序

### B. 检查每条陈述

### C. 修改错误设计

#### D. 查找容易误导用户的内容

● 阅读下列流程图：



当用判定覆盖法进行测试时，至少需要设计 (44) 个测试用例。

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

● WEB 应用链接测试不包括 (45)。

- (45) A. 无链接指向的页面      B. 错误的链接  
C. 客户端与服务器端的链接速率      D. 不存在的页面文件

● 在某大学学籍管理信息系统中，假设学生年龄的输入范围为 16~40，则根据黑盒测试中的等价类划分技术，下面划分正确的是 (46)。

- (46) A. 可划分为 2 个有效等价类，2 个无效等价类  
B. 可划分为 1 个有效等价类，2 个无效等价类  
C. 可划分为 2 个有效等价类，1 个无效等价类  
D. 可划分为 1 个有效等价类，1 个无效等价类

● 以下各项中，(47) 属于安装测试应关注的内容。

- ① 安装手册的评估    ② 安装选项和设置的测试  
③ 安装顺序测试    ④ 修复安装测试与卸载测试

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



● 下面关于软件测试的说法，(48)是错误的。

- (48) A. 软件测试就是程序测试  
B. 软件测试贯穿于软件定义和开发的整个期间  
C. 需求规格说明、设计规格说明都是软件测试的对象  
D. 程序是软件测试的对象

● 关于白盒测试与黑盒测试的最主要区别，正确的是(49)。

- (49) A. 白盒测试侧重于程序结构，黑盒测试侧重于功能  
B. 白盒测试可以使用测试工具，黑盒测试不能使用工具  
C. 白盒测试需要程序员参与，黑盒测试不需要  
D. 黑盒测试比白盒测试应用更广泛

● 软件测试按实施组织分，测试应该包括以下的(50)。

- ① 开发方测试 ② 用户方测试 ③ 第三方测试 ④ 验收测试 ⑤ 确认测试  
(50) A. ①、②、③ B. ③、④、⑤  
C. ①、②、④ D. ①、②、③、④、⑤

● 以下各项中，(51)属于需求说明书的评测内容。

- ①系统定义的目标是否与用户的要求一致  
②设计的约束条件或限制条件是否符合实际  
③是否考虑过软件需求的其他方案  
④软件的行为与它必须处理的信息、必须完成的功能是否一致  
(51) A. ①、②、④ B. ①、③、④  
C. ②、③、④ D. ①、②、③、④

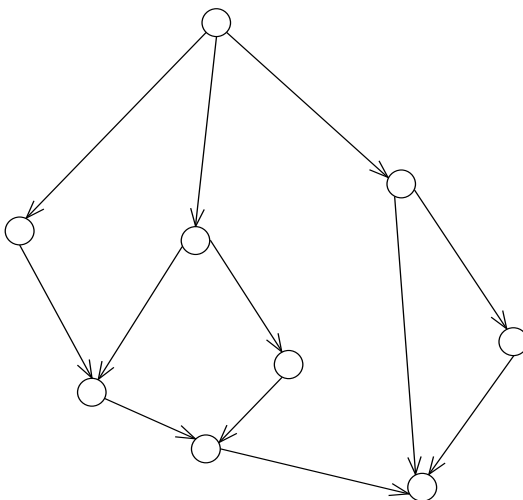
● 关于对第三方测试的描述，正确的观点是(52)。

- (52) A. 既不是用户，也不是开发人员所进行的测试就是第三方测试  
B. 第三方测试也称为独立测试，是由相对独立的组织进行的测试  
C. 第三方测试是在开发方与用户方的测试基础上进行的验证测试  
D. 第三方测试又被称为β测试

● 以下(53)属于在GB/T 18905《评价者用的过程》中定义软件评价过程的特性。

- ①可重复性 ②可再现性 ③公正性 ④客观性 ⑤合理性  
(53) A. ①、②、④ B. ①、②、③、④  
C. ③、④、⑤ D. ①、②、③、④、⑤

- 以下控制流程图的环路复杂性  $V(G)$  等于 (54)。



- (54) A. 4                      B. 5                      C. 6                      D. 1

- 通过疲劳强度测试，最容易发现 (55) 问题。

- (55) A. 并发用户数      B. 内存泄漏      C. 系统安全性      D. 功能错误

- 针对下列程序段，对于(A, B, C)的取值，以下 (56) 测试用例组合能够满足语句覆盖的要求。

IF ( ( A + 10 ) = 2 OR ( B - 20 ) < 3 ) THEN C = 0

IF ( ( A + 30 ) > 10 AND ( C - 30 ) < 0 ) THEN B = 30

- (56) A. (2, 30, 1)      B. (-20, 0, 30)      C. (-30, 20, 30)      D. (2, 20, 3)

- 针对下列程序段，对于(A, B)的取值，以下 (57) 测试用例组合能够满足条件覆盖的要求。

IF ( ( A - 10 ) = 20 AND ( B + 20 ) > 10 ) THEN C = 0

IF ( ( A - 30 ) < 10 AND ( B - 30 ) < 0 ) THEN B = 30

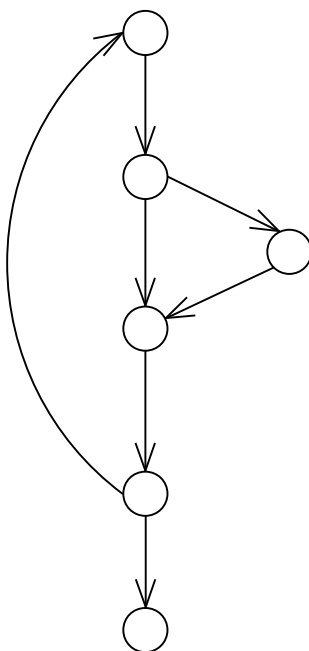
- ①A=50 B=-10      ②A=40 B=40      ③A=30 B=-10      ④A=30 B=30

- (57) A. ①②                      B. ③④                      C. ①④                      D. ②④

- 针对逻辑覆盖有下列叙述，(58) 是不正确的。

- (58) A. 达到 100%DC 要求就一定能够满足 100%SC 的要求  
 B. 达到 100%CC 要求就一定能够满足 100%SC 的要求  
 C. 达到 100%CDC 要求就一定能够满足 100%SC 的要求  
 D. 达到 100%MCDC 要求就一定能够满足 100%SC 的要求

- 以下所示程序控制流程图中有 (59) 条线性无关的基本路径。



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

- 下列叙述中，(60) 是正确的。

- (60) A. 白盒测试又称为逻辑驱动测试  
 B. 穷举路径测试可以查出程序中因遗漏路径而产生的错误  
 C. 一般而言，黑盒测试对结构的覆盖比白盒测试高  
 D. 必须根据软件需求说明文档生成用于白盒测试的测试用例

- 针对以下程序段，对于变量 c 的取值，至少需要 (61) 个测试用例才能够满足语句覆盖的要求。

```

c = ((u8_t *)q->payload)[i];
switch (c)
{
case SLIP_END:
    sio_send(SLIP_ESC, netif->state);
    sio_send(SLIP_ESC_END, netif->state);
    break;
case SLIP_ESC:
    sio_send(SLIP_ESC, netif->state);
    sio_send(SLIP_ESC_ESC, netif->state);
    break;
}
  
```

```

default:
    sio_send(c, netif->state);
    break;
}

```

(61) A. 4                      B. 3                      C. 2                      D. 1

● 针对以下 C 语言程序段，对于(MaxNum, Type)的取值，至少需要(62)个测试用例能够满足判定覆盖的要求。

```

while ( MaxNum-- > 0 )
{
    if ( 10 == Type )
        x = y * 2;
    else
        if ( 100 == Type )
            x = y + 10;
        else
            x = y - 20;
}

```

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

● 假设 A、B 为布尔变量，对于逻辑表达式 ( A && B )，至少需要(63)个测试用例才能完成 MCDC 覆盖。

(63) A. 4                      B. 3                      C. 2                      D. 1

● 广义的软件测试包括(64)。

- (64) A. 单元测试、集成测试、确认测试和系统测试  
 B. 确认、验证和测试  
 C. 需求评审、设计评审、单元测试和综合测试  
 D. 开发方测试、用户测试和第三方测试

● GB/T 16260 将软件的内部(外部)质量属性划分为六大质量特性，分别是(65)。

- (65) A. 功能性，可靠性，易用性，效率，维护性和可移植性  
 B. 功能性、可靠性、易用性、效率、稳定性和可移植性  
 C. 功能性、可靠性、安全性、效率、易用性和可移植性  
 D. 功能性、可靠性、兼容性、效率、稳定性和可移植性

● 软件内部/外部质量模型中，以下(66)不是功能性包括的子特性。

- (66) A. 适合性              B. 准确性              C. 稳定性              D. 互操作性

● 《GB/T 18905 软件工程 产品评价》中确定的通用评价过程包括四个方面，其中有关“规定评价”部分包含的内容有(67)。

- (67) A. 选择度量、建立度量评定等级、确立评估准则
- B. 指定质量模型、选择度量、建立度量评定等级
- C. 选择度量、建立度量评定等级、制定评价计划
- D. 确定产品类型、选择度量、建立度量评定等级

● 下列测试工具中，使用(68)执行自动化负载压力测试，使用(69)执行代码静态结构分析，使用(70)执行网络测试。

- |                            |               |
|----------------------------|---------------|
| (68) A. SmartBits          | B. Logiscope  |
| C. Quick Test Professional | D. LoadRunner |
| (69) A. SmartBits          | B. Logiscope  |
| C. Quick Test Professional | D. LoadRunner |
| (70) A. SmartBits          | B. Logiscope  |
| C. Quick Test Professional | D. LoadRunner |

● (71) analysis emphasizes the drawing of pictorial system models to document and validate both existing and/or proposed systems. Ultimately, the system models become the (72) for designing and constructing an improved system. (73) is such a technique. The emphasis in this technique is process-centered. Systems analysts draw a series of process models called (74). (75) is another such technique that integrates data and process concerns into constructs called objects.

- |                             |                             |                 |              |
|-----------------------------|-----------------------------|-----------------|--------------|
| (71) A. Prototyping         | B. Accelerated              | C. Model-driven | D. Iterative |
| (72) A. image               | B. picture                  | C. layout       | D. blueprint |
| (73) A. Structured analysis | B. Information Engineering  |                 |              |
| C. Discovery Prototyping    | D. Object-Oriented analysis |                 |              |
| (74) A. PERT                | B. DFD                      | C. ERD          | D. UML       |
| (75) A. Structured analysis | B. Information Engineering  |                 |              |
| C. Discovery Prototyping    | D. Object-Oriented analysis |                 |              |