

全国计算机技术与软件专业技术资格（水平）考试

2008 年下半年 软件设计师 上午试卷（B）

（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

● 2008 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. 11 B. 12 C. 10 D. 9

（89） A. 18 B. 19 C. 20 D. 21

因为考试日期是“12 月 21 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）。

● 计算机内存一般分为静态数据区、代码区、栈区和堆区，若某指令的操作数之一采用立即数寻址方式，则该操作数位于(1)。

- (1) A. 静态数据区 B. 代码区 C. 栈区 D. 堆区

● 计算机在进行浮点数的相加（减）运算之前先进行对阶操作，若 x 的阶码大于 y 的阶码，则应将(2)。

- (2) A. x 的阶码缩小至与 y 的阶码相同，且使 x 的尾数部分进行算术左移
B. x 的阶码缩小至与 y 的阶码相同，且使 x 的尾数部分进行算术右移
C. y 的阶码扩大至与 x 的阶码相同，且使 y 的尾数部分进行算术左移
D. y 的阶码扩大至与 x 的阶码相同，且使 y 的尾数部分进行算术右移

● 在 CPU 中，(3) 可用于传送和暂存用户数据，为 ALU 执行算术逻辑运算提供工作区。

- (3) A. 程序计数器 B. 累加寄存器 C. 程序状态寄存器 D. 地址寄存器

● 下面关于在 I/O 设备与主机间交换数据的叙述，(4) 是错误的。

- (4) A. 中断方式下，CPU 需要执行程序来实现数据传送任务
B. 中断方式和 DMA 方式下，CPU 与 I/O 设备都可同步工作
C. 中断方式和 DMA 方式中，快速 I/O 设备更适合采用中断方式传递数据
D. 若同时接到 DMA 请求和中断请求，CPU 优先响应 DMA 请求

● 下面关于校验方法的叙述，(5) 是正确的。

- (5) A. 采用奇偶校验可检测数据传输过程中出现一位数据错误的位置并加以纠正
B. 采用海明校验可检测数据传输过程中出现一位数据错误的位置并加以纠正
C. 采用海明校验，校验码的长度和位置可随机设定
D. 采用 CRC 校验，需要将校验码分散开并插入数据的指定位置中

● Cache 用于存放主存数据的部分拷贝，主存单元地址与 Cache 单元地址之间的转换工作由(6) 完成。

- (6) A. 硬件 B. 软件 C. 用户 D. 程序员

● 在 Windows Server 2003 下若选择安全登录，则首先需要按(7) 组合键。

- (7) A. Shift+Alt+Esc B. Ctrl+Alt+Tab C. Ctrl+Shift D. Ctrl+Alt+Del

● 为了防止电子邮件中的恶意代码，应该用(8) 方式阅读电子邮件。

- (8) A. 文本 B. 网页 C. 程序 D. 会话

● TCP/IP 在多个层引入了安全机制，其中 TLS 协议位于(9)。

- (9) A. 数据链路层 B. 网络层 C. 传输层 D. 应用层

● 我国专利申请的原则之一是(10)。

- (10) A. 申请在先 B. 申请在先与使用在先相结合
C. 使用在先 D. 申请在先、使用在先或者二者结合

● 李某在《电脑与编程》杂志上看到张某发表的一组程序，颇为欣赏，就复印了一百份作为程序设计辅导材料发给了学生。李某又将这组程序逐段加以评析，写成评论文章后投到《电脑编程技巧》杂志上发表。李某的行为(11)。

- (11) A. 侵犯了张某的著作权，因为其未经许可，擅自复印张某的程序
B. 侵犯了张某的著作权，因为在评论文章中全文引用了发表的程序
C. 不侵犯张某的著作权，其行为属于合理使用
D. 侵犯了张某的著作权，因为其擅自复印，又在其发表的文章中全文引用了张某的程序

● MP3 是目前最流行的数字音乐压缩编码格式之一，其命名中“MP”是指(12)，“3”是指(13)。

- (12) A. media player B. multiple parts C. music player D. MPEG-1 Audio
(13) A. MPEG-3 B. version 3 C. part 3 D. layer 3

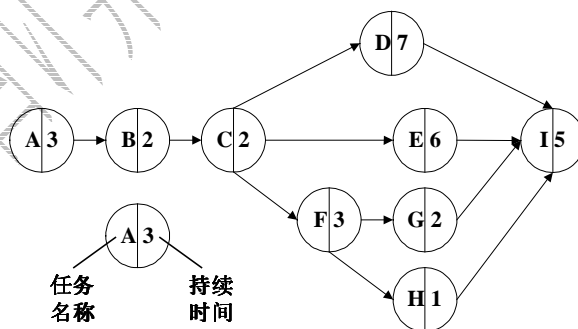
● 某数码相机内置 128MB 的存储空间，拍摄分辨率设定为 1600×1200 像素，颜色深度为 24 位，若不采用压缩存储技术，使用内部存储器最多可以存储(14)张照片。

- (14) A. 12 B. 22 C. 13 D. 23

● 在面向对象系统中，用(15)关系表示一个较大的“整体”类包含一个或多个较小的“部分”类。

- (15) A. 泛化 B. 聚合 C. 概化 D. 合成

● 若一个项目由 9 个主要任务构成，其计划图（如下图所示）展示了任务之间的前后关系以及每个任务所需天数，该项目的关键路径是(16)，完成项目所需的最短时间是(17)天。



- (16) A. $A \rightarrow B \rightarrow C \rightarrow D \rightarrow I$ B. $A \rightarrow B \rightarrow C \rightarrow E \rightarrow I$
C. $A \rightarrow B \rightarrow C \rightarrow F \rightarrow G \rightarrow I$ D. $A \rightarrow B \rightarrow C \rightarrow F \rightarrow H \rightarrow I$
(17) A. 16 B. 17 C. 18 D. 19

● RUP (Rational Unified Process) 分为 4 个阶段，每个阶段结束时都有重要的里程碑，其中生命周期架构是在 (18) 结束时的里程碑。

- (18) A. 初启阶段 B. 精化阶段 C. 构建阶段 D. 移交阶段

● 在软件工程环境中进行风险识别时，常见的、已知的及可预测的风险类包括产品规模、商业影响等，与开发工具的可用性及质量相关的风险是 (19)。

- (19) A. 客户特性 B. 过程定义 C. 开发环境 D. 构建技术

● 编译程序对高级语言源程序进行翻译时，需要在该程序的地址空间中为变量指定地址，这种地址称为 (20)。

- (20) A. 逻辑地址 B. 物理地址 C. 接口地址 D. 线性地址

● 程序设计语言一般都提供多种循环语句，例如实现先判断循环条件再执行循环体的 while 语句和先执行循环体再判断循环条件的 do-while 语句。关于这两种循环语句，在不改变循环体的条件下， (21) 是正确的。

- (21) A. while 语句的功能可由 do-while 语句实现
B. do-while 语句的功能可由 while 语句实现
C. 若已知循环体的次数，则只能使用 while 语句
D. 循环条件相同时，do-while 语句的执行效率更高

● 表达式 $(a-b)*(c+5)$ 的后缀式是 (22)。

- (22) A. $a\ b\ c\ 5\ +\ * -$ B. $a\ b - c + 5 *$
C. $a\ b\ c - * 5 +$ D. $a\ b - c\ 5 + *$

● 假设系统中有四类互斥资源 R1、R2、R3 和 R4，可用资源数分别为 9、6、3 和 3。在 T_0 时刻系统中有 P1、P2、P3 和 P4 四个进程，这些进程对资源的最大需求量和已分配资源数如下表所示。在 T_0 时刻系统剩余的可用资源数分别为 (23)。如果 P1、P2、P3 和 P4 进程按 (24) 序列执行，那么系统状态是安全的。

资源 进程	最大需求量				已分配资源数			
	R1	R2	R3	R4	R1	R2	R3	R4
P1	6	4	2	1	1	1	1	1
P2	2	2	2	1	2	1	1	1
P3	8	1	1	1	2	1	0	0
P4	2	2	1	1	1	2	1	1

- (23) A. 2、1、0 和 1 B. 3、1、0 和 0

- C. 3、1、1 和 1 D. 3、0、1 和 1

- (24) A. P1→P2→P4→P3 B. P2→P1→P4→P3

- C. P3→P4→P1→P2 D. P4→P2→P1→P3

● 某文件管理系统为了记录磁盘的使用情况，在磁盘上建立了位示图(bitmap)。若系统中字长为 16 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 (25) 个字中描述。

- (25) A. 256 B. 257 C. 512 D. 513

● 在操作系统设备管理中，通常临界资源不能采用 (26) 分配算法。

- (26) A. 静态优先级 B. 动态优先级 C. 时间片轮转 D. 先来先服务

● 某虚拟存储系统采用最近最少使用 (LRU) 页面淘汰算法。假定系统为每个作业分配 3 个页面的主存空间，其中一个页面用来存放程序。现有某作业的部分语句如下：

```
Var A: Array[1..128,1..128] OF integer;
```

```
  i,j: integer;
```

```
  FOR i:=1 to 128 DO
```

```
    FOR j:=1 to 128 DO
```

```
      A[i,j]:=0;
```

设每个页面可存放 128 个整数变量，变量 i、j 放在程序页中，矩阵 A 按行序存放。初始时，程序及变量 i、j 已在内存，其余两页为空。在上述程序片段执行过程中，共产生 (27) 次缺页中断。最后留在内存中的是矩阵 A 的最后 (28) 。

- (27) A. 64 B. 128 C. 256 D. 512

- (28) A. 2 行 B. 2 列 C. 1 行 D. 1 列

● 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初始级、可重复级、定义级、管理级和优化级。其中 (29) 对软件过程和产品都有定量的理解与控制。

- (29) A. 可重复级和定义级 B. 定义级和管理级
 C. 管理级和优化级 D. 定义级、管理级和优化级

● 极限编程 (XP) 包含了策划、设计、编程和测试四个活动，其十二个最佳实践中的“持续集成”实践在 (30) 活动中进行。

- (30) A. 策划和设计 B. 设计和编程
 C. 设计和测试 D. 编程和测试

● ISO/IEC 9126 软件质量模型中第一层定义了六个质量特性，并为各质量特性定义了相应的质量子特性。子特性 (31) 属于可靠性质量特性。

- (31) A. 准确性 B. 易理解性 C. 成熟性 D. 易学性

● UP (统一过程) 是用例驱动的、以架构为核心、迭代和增量的软件过程框架，它提供了一种 (32) 的特性。

- (32) A. 演进 B. 敏捷 C. 测试驱动 D. 持续集成

● 系统测试人员与系统开发人员需要通过文档进行沟通，系统测试人员应根据一系列文档对系统进行测试，然后将工作结果撰写成(33)，交给系统开发人员。

- (33) A. 系统开发合同 B. 系统设计说明书
C. 测试计划 D. 系统测试报告

● 系统的可维护性可以用系统的可维护性评价指标来衡量。系统的可维护性评价指标不包括(34)。

- (34) A. 可理解性 B. 可修改性 C. 准确性 D. 可测试性

● 某程序根据输入的三条线段长度，判断这三条线段能否构成三角形。以下 6 个测试用例中，(35) 两个用例属于同一个等价类。

- ①6、7、13； ②4、7、10； ③9、20、35；
④9、11、21； ⑤5、5、4； ⑥4、4、4。
(35) A. ①② B. ③④ C. ⑤⑥ D. ①④

● 在模拟环境下，常采用黑盒测试检验所开发的软件是否与需求规格说明书一致。其中有效性测试属于(36)中的一个步骤。

- (36) A. 单元测试 B. 集成测试 C. 确认测试 D. 系统测试

● 面向对象分析与设计中的(37)是指一个模块在扩展性方面应该是开放的，而在更改性方面应该是封闭的；而(38)是指子类应当可以替换父类并出现在父类能够出现的任何地方。

- (37) A. 开闭原则 B. 替换原则 C. 依赖原则 D. 单一职责原则
(38) A. 开闭原则 B. 替换原则 C. 依赖原则 D. 单一职责原则

● 在选择某种面向对象语言进行软件开发时，不需要着重考虑的因素是，该语言(39)。

- (39) A. 将来是否能够占据市场主导地位
B. 类库是否丰富
C. 开发环境是否成熟
D. 是否支持全局变量和全局函数的定义

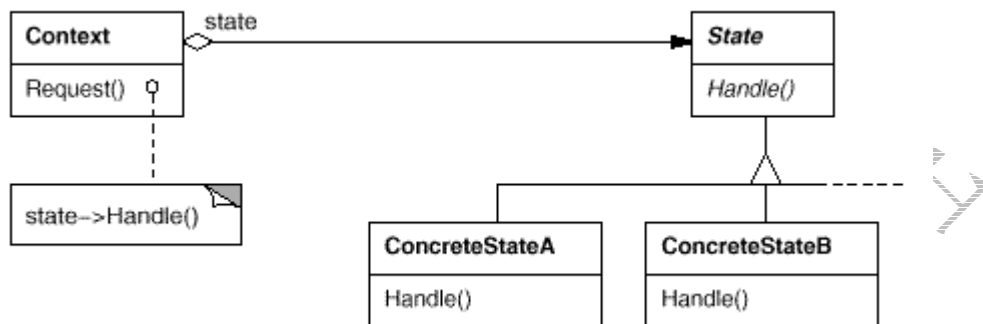
● (40)限制了创建类的实例数量，而(41)将一个类的接口转换成客户希望的另外一个接口，使得原本由于接口不兼容而不能一起工作的那些类可以一起工作。

- (40) A. 命令模式 (Command) B. 适配器模式 (Adapter)
C. 策略模式 (Strategy) D. 单例模式 (Singleton)
(41) A. 命令模式 (Command) B. 适配器模式 (Adapter)
C. 策略模式 (Strategy) D. 单例模式 (Singleton)

● (42) 是指在运行时把过程调用和响应调用所需要执行的代码加以结合。

(42) A. 绑定 B. 静态绑定 C. 动态绑定 D. 继承

● (43) 设计模式允许一个对象在其内部状态改变时改变它的行为。下图为这种设计模式的类图，已知类 **State** 为抽象类，则类 (44) 的实例代表了 **Context** 对象的状态。



(43) A. 单件 (Singleton) B. 桥接 (Bridge)

C. 组合 (Composite) D. 状态 (State)

(44) A. Context B. concreteStateA

C. Handle D. State

● 在 UML 的各种视图中，(45) 显示外部参与者观察到的系统功能；(46) 从系统的静态结构和动态行为角度显示系统内部如何实现系统的功能；(47) 显示的是源代码以及实际执行代码的组织结构。

(45) A. 用例视图 B. 进程视图 C. 实现视图 D. 逻辑视图

(46) A. 用例视图 B. 进程视图 C. 实现视图 D. 逻辑视图

(47) A. 用例视图 B. 进程视图 C. 实现视图 D. 逻辑视图

● 给定文法 $G[S]$ 及其非终结符 **A**， $FIRST(A)$ 定义为：从 **A** 出发能推导出的终结符号的集合 (**S** 是文法的起始符号，为非终结符)。对于文法 $G[S]$ ：

$S \rightarrow [L] | a$

$L \rightarrow L, S | S$

其中， $G[S]$ 包含的四个终结符号分别为：

a , $[$ $]$

则 $FIRST(S)$ 的成员包括 (48)。

(48) A. a B. a 、 $[$ C. a 、 $[$ 和 $]$ D. a 、 $[$ 、 $]$ 和 $,$

● 高级语言源程序的编译过程分若干个阶段，分配寄存器属于 (49) 阶段的工作。

(49) A. 词法分析 B. 语法分析 C. 语义分析 D. 代码生成

● 设某上下文无关文法如下： $S \rightarrow 11 \mid 1001 \mid S0 \mid SS$ ，则该文法所产生的所有二进制字符串都具有的特点是 (50)。

- (50) A. 能被 3 整除
B. 0、1 出现的次数相等
C. 0 和 1 的出现次数都为偶数
D. 能被 2 整除

● 某学校学生、教师和课程实体对应的关系模式如下：

学生（学号，姓名，性别，年龄，家庭住址，电话）

课程（课程号，课程名）

教师（职工号，姓名，年龄，家庭住址，电话）

如果一个学生可以选修多门课程，一门课程可以有多个学生选修；一个教师只能讲授一门课程，但一门课程可以有多个教师讲授。由于学生和课程之间是一个 (51) 的联系，所以 (52)。又由于教师和课程之间是一个 (53) 的联系，所以 (54)。

- (51) A. 1 对 1 B. 1 对多 C. 多对 1 D. 多对多
(52) A. 不需要增加一个新的关系模式
B. 不需要增加一个新的关系模式，只需要将 1 端的码插入多端
C. 需要增加一个新的选课关系模式，该模式的主键应该为课程号
D. 需要增加一个新的选课关系模式，该模式的主键应该为课程号和学号
(53) A. 1 对 1 B. 1 对多 C. 多对 1 D. 多对多
(54) A. 不需要增加一个新的关系模式，只需要将职工号插入课程关系模式
B. 不需要增加一个新的关系模式，只需要将课程号插入教师关系模式
C. 需要增加一个新的选课关系模式，该模式的主键应该为课程号
D. 需要增加一个新的选课关系模式，该模式的主键应该为课程号和教师号

● 关系 R、S 如下图所示，关系代数表达式 $\pi_{1,5,6}(\sigma_{2=5}(R \times S)) = \underline{(55)}$ ，该表达式与 (56) 等价。

A	B	C
3	0	3
2	5	6
5	8	9
8	11	12

R

A	B	C
3	10	11
4	11	6
5	10	13
6	11	14

S

(55) A.

A	B	C
3	0	3
5	8	9

B.

A	B	C
8	11	6
8	11	14

C.

A	B	C
5	10	11
5	10	13

D.

A	B	C
2	11	6
2	11	14

- (56) A. $\pi_{A,B,C}(\sigma_{B=B}(R \times S))$
B. $\pi_{R.A,R.B,R.C}(\sigma_{R.B=S.B}(R \times S))$
C. $\pi_{R.A,S.B,S.C}(\sigma_{R.B=S.B}(R \times S))$
D. $\pi_{R.A,S.B,S.C}(\sigma_{R.B=S.C}(R \times S))$

● 一个具有 m 个结点的二叉树，其二叉链表结点（左、右孩子指针分别用 **left** 和 **right** 表示）中的空指针总数必定为 (57) 个。为形成中序（先序、后序）线索二叉树，现对该二叉链表所有结点进行如下操作：若结点 p 的左孩子指针为空，则将该左指针改为指向 p 在中序（先序、后序）遍历序列的前驱结点；若 p 的右孩子指针为空，则将该右指针改为指向 p 在中序（先序、后序）遍历序列的后继结点。假设指针 s 指向中序（先序、后序）线索二叉树中的某结点，则 (58)。

(57) A. $m+2$ B. $m+1$ C. m D. $m-1$

(58) A. $s \rightarrow \text{right}$ 指向的结点一定是 s 所指结点的直接后继结点

B. $s \rightarrow \text{left}$ 指向的结点一定是 s 所指结点的直接前驱结点

C. 从 s 所指结点出发的 **right** 链可能构成环

D. s 所指结点的 **left** 和 **right** 指针一定指向不同的结点

● (59) 的邻接矩阵是一个对称矩阵。

(59) A. 无向图 B. AOV 网 C. AOE 网 D. 有向图

● 将一个无序序列中的元素依次插入到一棵 (60)，并进行中序遍历，可得到一个有序序列。

(60) A. 完全二叉树 B. 最小生成树 C. 二叉排序树 D. 最优二叉树

● 广义表中的元素可以是原子，也可以是表，因此广义表的适用存储结构是 (61)。

(61) A. 链表 B. 静态数组 C. 动态数组 D. 散列表

● 某一维数组中依次存放了数据元素 12,23,30,38,41,52,54,76,85，在用折半（二分）查找方法（向上取整）查找元素 54 时，所经历“比较”运算的数据元素依次为 (62)。

(62) A. 41, 52, 54 B. 41, 76, 54 C. 41, 76, 52, 54 D. 41, 30, 76, 54

● 具有 n 个顶点、 e 条边的图采用邻接表存储结构，进行深度优先遍历和广度优先遍历运算的时间复杂度均为 (63)。

(63) A. $O(n^2)$ B. $O(e^2)$ C. $O(n*e)$ D. $O(n+e)$

● 给定一组长度为 n 的无序序列，将其存储在一维数组 $a[0..n-1]$ 中。现采用如下方法找出其中的最大元素和最小元素：比较 $a[0]$ 和 $a[n-1]$ ，若 $a[0]$ 较大，则将二者的值进行交换；再比较 $a[1]$ 和 $a[n-2]$ ，若 $a[1]$ 较大，则交换二者的值；然后依次比较 $a[2]$ 和 $a[n-3]$ 、 $a[3]$ 和 $a[n-4]$ 、...，使得每一对元素中的较小者被交换到低下标端。重复上述方法，在数组的前 $n/2$ 个元素中查找最小元素，在后 $n/2$ 个元素查找最大元素，从而得到整个序列的最小元素和最大元素。上述方法采用的算法设计策略是 (64)。

(64) A. 动态规划法 B. 贪心法 C. 分治法 D. 回溯法

● 设某算法的计算时间表示为递推关系式 $T(n) = T(n-1) + n$ ($n > 0$) 及 $T(0) = 1$, 则该算法的时间复杂度为 (65)。

(65) A. $O(\lg n)$ B. $O(n \lg n)$ C. $O(n)$ D. $O(n^2)$

● ADSL 是一种宽带接入技术, 这种技术使用的传输介质是 (66)。

(66) A. 电话线 B. CATV 电缆
C. 基带同轴电缆 D. 无线通信网

● 下面关于网络系统设计原则的论述, 正确的是 (67)。

(67) A. 应尽量采用先进的网络设备, 获得最高的网络性能
B. 网络总体设计过程中, 只需要考虑近期目标即可, 不需要考虑扩展性
C. 网络系统应采用开放的标准和技术
D. 网络需求分析独立于应用系统的需求分析

● 下面的选项中, 属于本地回路地址的是 (68)。

(68) A. 120.168.10.1 B. 10.128.10.1
C. 127.0.0.1 D. 172.16.0.1

● 下面关于防火墙的说法, 正确的是 (69)。

(69) A. 防火墙一般由软件以及支持该软件运行的硬件系统构成
B. 防火墙只能防止未经授权的信息发送到内网
C. 防火墙能准确地检测出攻击来自哪一台计算机
D. 防火墙的主要支撑技术是加密技术

● Internet 上的 DNS 服务器中保存有 (70)。

(70) A. 主机名 B. 域名到 IP 地址的映射表
C. 所有主机的 MAC 地址 D. 路由表

● It should go without saying that the focus of UML is modeling. However, what that means, exactly, can be an open-ended question. (71) is a means to capture ideas, relationships, decisions, and requirements in a well-defined notation that can be applied to many different domains. Modeling not only means different things to different people, but also it can use different pieces of UML depending on what you are trying to convey. In general, a UML model is made up of one or more (72). A diagram graphically represents things, and the relationships between these things. These (73) can be representations of real-world objects, pure software constructs, or a description of the behavior of some other objects. It is common for an individual thing to show up on multiple diagrams; each diagram represents a particular interest, or view, of the thing being modeled. UML 2.0 divides diagrams into two categories:

structural diagrams and behavioral diagrams. (74) are used to capture the physical organization of the things in your system, i.e., how one object relates to another. (75) focus on the behavior of elements in a system. For example, you can use behavioral diagrams to capture requirements, operations, and internal state changes for elements.

- | | | | | |
|------|------------------------|--------------|------------------------|-----------------------|
| (71) | A. Programming | B. Analyzing | C. Designing | D. Modeling |
| (72) | A. views | B. diagrams | C. user views | D. structure pictures |
| (73) | A. things | B. pictures | C. languages | D. diagrams |
| (74) | A. Activity diagrams | | B. Use-case diagrams | |
| | C. Structural diagrams | | D. Behavioral diagrams | |
| (75) | A. Activity diagrams | | B. Use-case diagrams | |
| | C. Structural diagrams | | D. Behavioral diagrams | |