

参考答案

一、单项选择题：

1. A 2. A 4. B 5. B 6. C 7. C 8. C 9. B 10. B 11. B 12. A 13. C 14. A 15. C
16. B 17. A 18. C 19. C 20. A 21. B 22. C 23. A 24. C 25. A 26. C 27. B 29. C 30. C
31. C 32. B 34. B 35. C 36. C 38. A 39. B 40. C 41. B 42. A 43. C 44. B 45. A

二、填空题：

- | | |
|-----------------------|------------------------|
| 1. 聚集 组合 | 2. 描述计算机程序的文档 数据 基线 |
| 3. 累计经济效益(折合成现在值) | 4. 文档 |
| 5. 驱动模块 桩模块 | 6. A-0 3-6 |
| 7. 逻辑 | 8. 源程序 标号交叉引用表 |
| 9. 输入流 输出流 | 10. 50 |
| 11. $E - N + 2$ | 12. 聚合 |
| 13. 深度优先 广度优先 | 14. 环形复杂性 以前没有处理的语句或条件 |
| 15. 深度 宽度 | 16. 依赖 越低 |
| 17. 大大压缩简化了的系统分析和设计过程 | 18. 确定系统必须完成哪些工作 |
| 19. 结构化 | 20. 继承 对象组合 |
| 21. 输入数据 预期输出结果 | 22. 越大 |
| 23. 主要参与者 泛化 | 24. 过程 |
| 25. 变化 | 26. 过程实体 存取限制 |
| 27. UML 元模型 | 28. 判定表达式中每个条件的各种可能的值 |
| 29. 分解与抽象 功能 | 30. ISO9001 |
| 31. 风险发生的可能性 | |

三、名词解释：

1. **集成测试**也称组装测试或联调。是指在单元测试的基础上，将所有模块按照概要设计要求组装成一个完整的系统而进行的测试。模块组装的方式有两种：渐增式和非渐增式。
2. **设计模式**是普通面向对象设计问题的解决方案，这类问题以一组交互类的形式出现，用户根据需要定制这些交互类以形成专门的设计。所有设计模式均可以从四个方面加以描述：模式名称、问题描述、解决方案和效果。
3. **信息隐藏**是指在模块设计时，使得一个模块内包含的信息(过程或数据)，对于不需要这些信息的其他模块来说是不能访问的。这样，改变模块的内部过程和数据的设计不会影响到其它模块。
4. **用例**是参与者为达到某个目的而与系统进行的一系列交互，执行结果将为参与者提供可度量的价值。从参与者的角度来看，用例应该是一个完整的任务，在一个相对较短的时间内完成。
5. **结构化方法**是一种传统的软件开发方法，其总的指导思想是自顶向下、初步求精、单入口、单出口，其基本原则是抽象和功能分解。
6. **MVC**即模型、视图、控制器。MVC是Model—View—Controller的简写。“Model”代表的是应用的业务逻辑，模型是应用对象，其包含一些数据值，维护数据并提供数据的访问方法。“View”是应用的表示层，视图显示模型的部分或数据，多个视图可以以不同的方式来显示模型中数据。“Controller”是提供应用的处理过程控制，处理用户输入事件。通过MVC把应用逻辑，处理过程和显示逻辑分成不同的组件实现。

四、判断题：

1. × 2. × 3. × 4. √ 5. √ 6. × 7. √ 8. √ 9. × 10. × 11. × 12. × 13. √ 14. √ 15. √
16. × 17. √ 18. × 19. × 20. × 21. √ 22. ×

五、简答题：

1. 在需求分析和设计阶段建立原型的好处有：
 - (1) 增进开发人员和用户对系统需求的一致理解，明确软件需求，尤其是功能含糊的需求。
 - (2) 原型提供了一种有力的学习手段，能够使用户尽早接触系统。
 - (3) 原型可以用来识别或化解风险。
 - (4) 原型有的可以原封不动地成为产品，有的略加修改就可以成为最终系统的一个组成部分，这样有

利于建成最终系统。

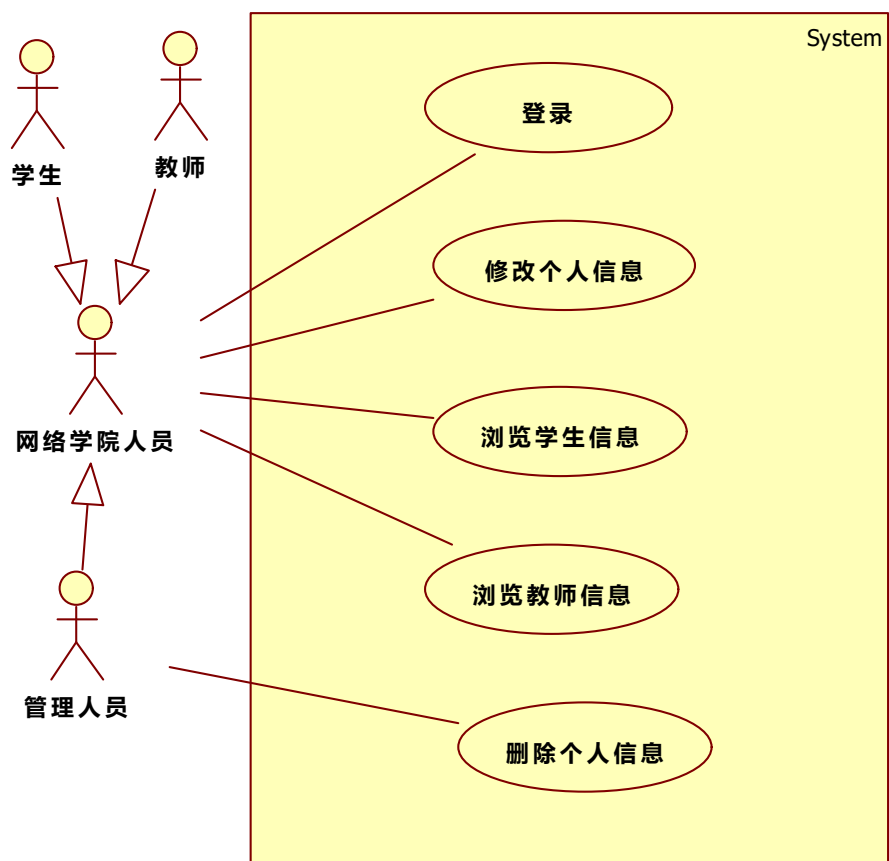
2. 扇入则定义为一个模块被调用（或控制）的模块个数；扇出表示一个模块直接调用（或控制）的其他模块数目。
3. UML 的特点是：
 - （1）统一了面向对象方法的有关概念和描述方法；
 - （2）表达能力强，能对各种并发分布式系统进行建模，且 UML 还提供了扩展机制；
 - （3）UML 专注于一种标准的建模语言，而不是一个开发过程。

教材中

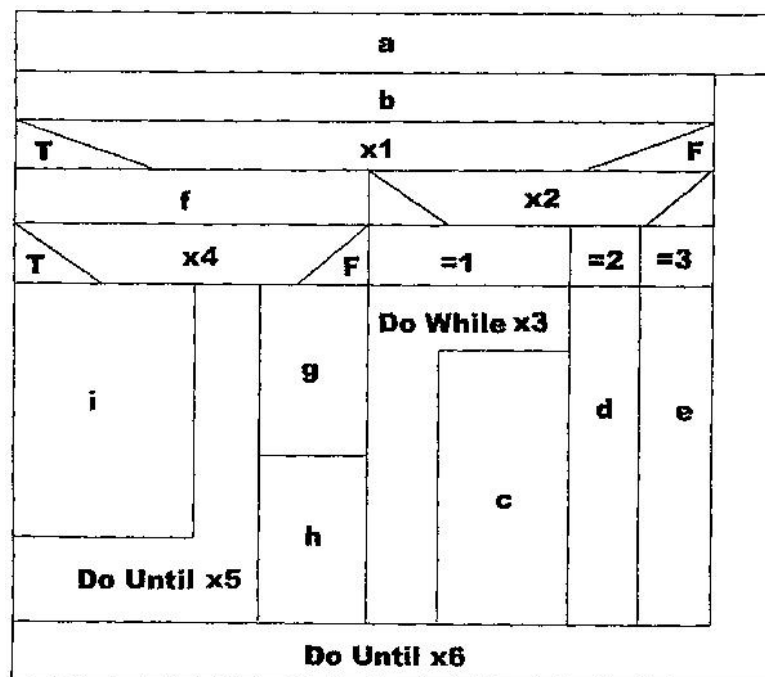
5. 软件测试并不等于程序代码测试，其应该贯穿于软件定义与开发的整个期间。因此，需求分析、概要设计以及程序编码等各阶段所得的文档，包括需求规格说明、概要设计说明、详细设计规格说明以及源程序代码都应成为软件测试的对象。
6. 在软件工程中，文档的作用是：
 - （1）提高软件开发过程的能见度；
 - （2）记录开发过程的有关信息便于使用与维护；
 - （3）作为开发人员阶段工作成果和结束标志；
 - （4）提高开发效率；
 - （5）提供软件运行、维护和培训有关资料；
 - （6）便于用户了解软件功能、性能。
7. 因为如果开发人员和用户属于不同的组织机构，则一般在正式的软件开发之前均需要签订合同，所有的风险分析均必须在签订合同之前完成。但螺旋模型需要在软件开发的每一个阶段都进行风险分析，当重要风险不能解决时，项目应该终止。由此螺旋模型可能因为风险使得合同在完成之前就终止，从而导致法律诉讼。故螺旋模型不适合开发人员和用户属于不同组织机构的情况。
8. 软件测试不应仅作为一个独立的软件开发阶段，而应贯穿到软件开发的各个阶段中。通过各阶段的评审，在早期把错误解决，以减少错误放大效应，这样不仅可以提高软件质量，而且也是降低软件成本的一个重要措施。由此，软件测试应该尽早和不断地进行。
12. 根据原型机的不同作用，有三类原型模型：
 - （1）探索型原型。这种类型的原型模型是把原型用于开发的需求分析阶段，目的是要弄清用户的需求，确定所期望的特性，并探索各种方案的可行性。它主要针对开发目标模糊，用户与开发着对项目都缺乏经验的情况，通过对原型的开发来明确用户的需求。
 - （2）实验型原型。这种原型主要用于设计阶段，考核实现方案是否合适，能否实现，对于一个大型系统，若对设计方案心中没有把握时，可通过这种原型来证实设计方案的正确性。
 - （3）演化型原型。这种原型主要用于及早向用户提交一个原型系统，该原型系统或者包含系统的框或者包含系统的主要功能。在得到用户的认可后，将原型系统不断扩充演变为最终的软件系统，它将原型的思路扩展到软件开发的全过程。

六、综合题：

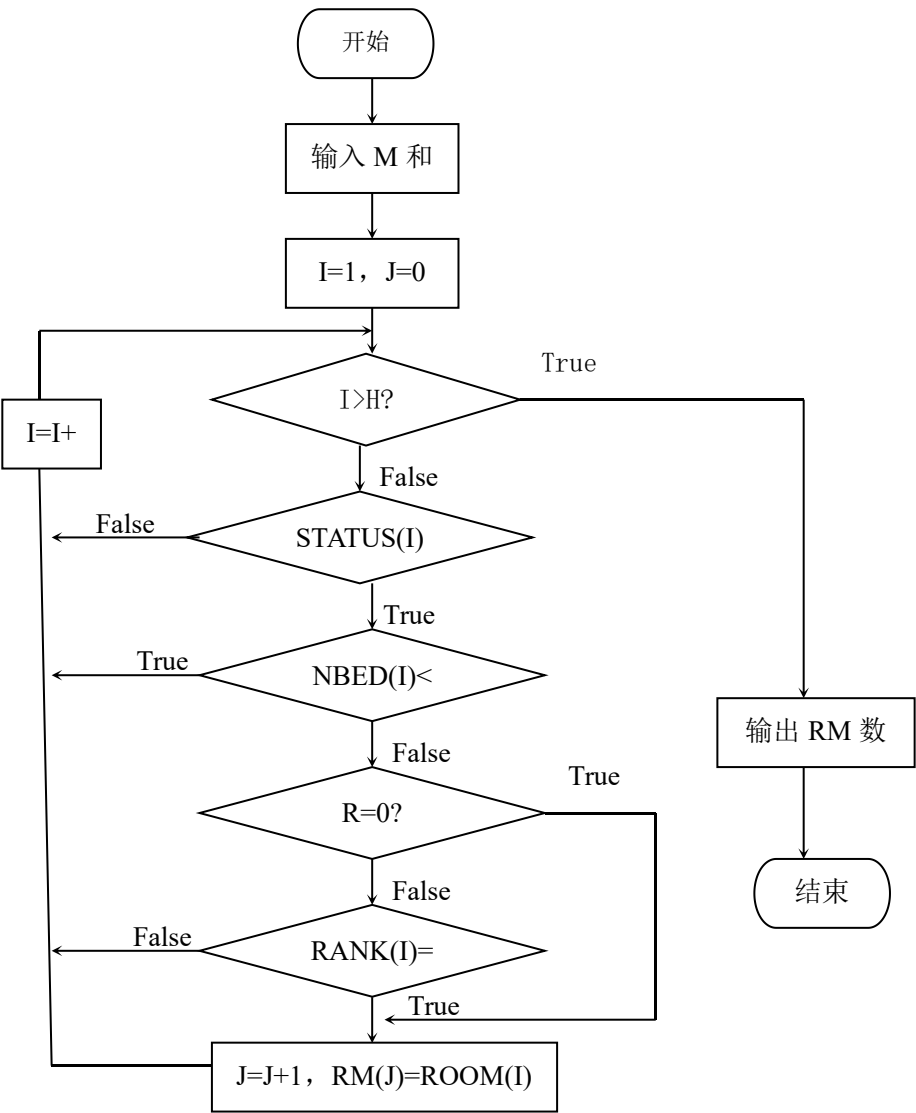
- 1.



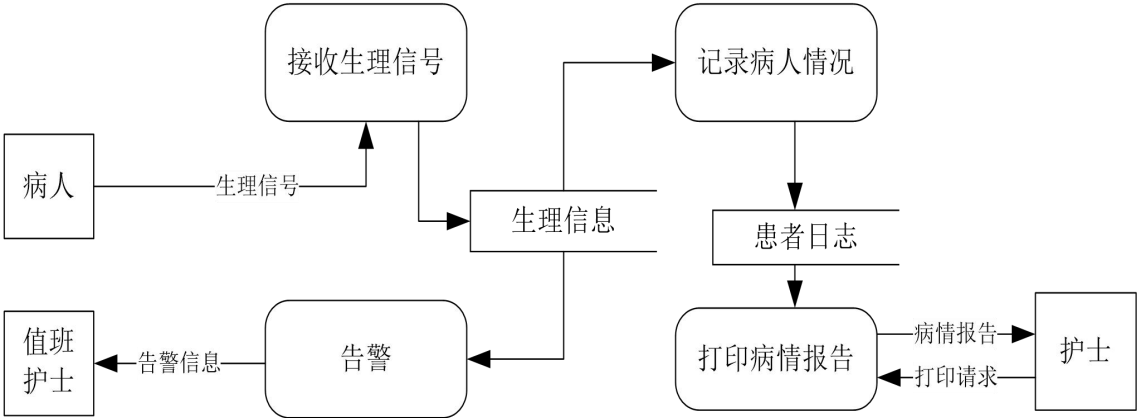
2.



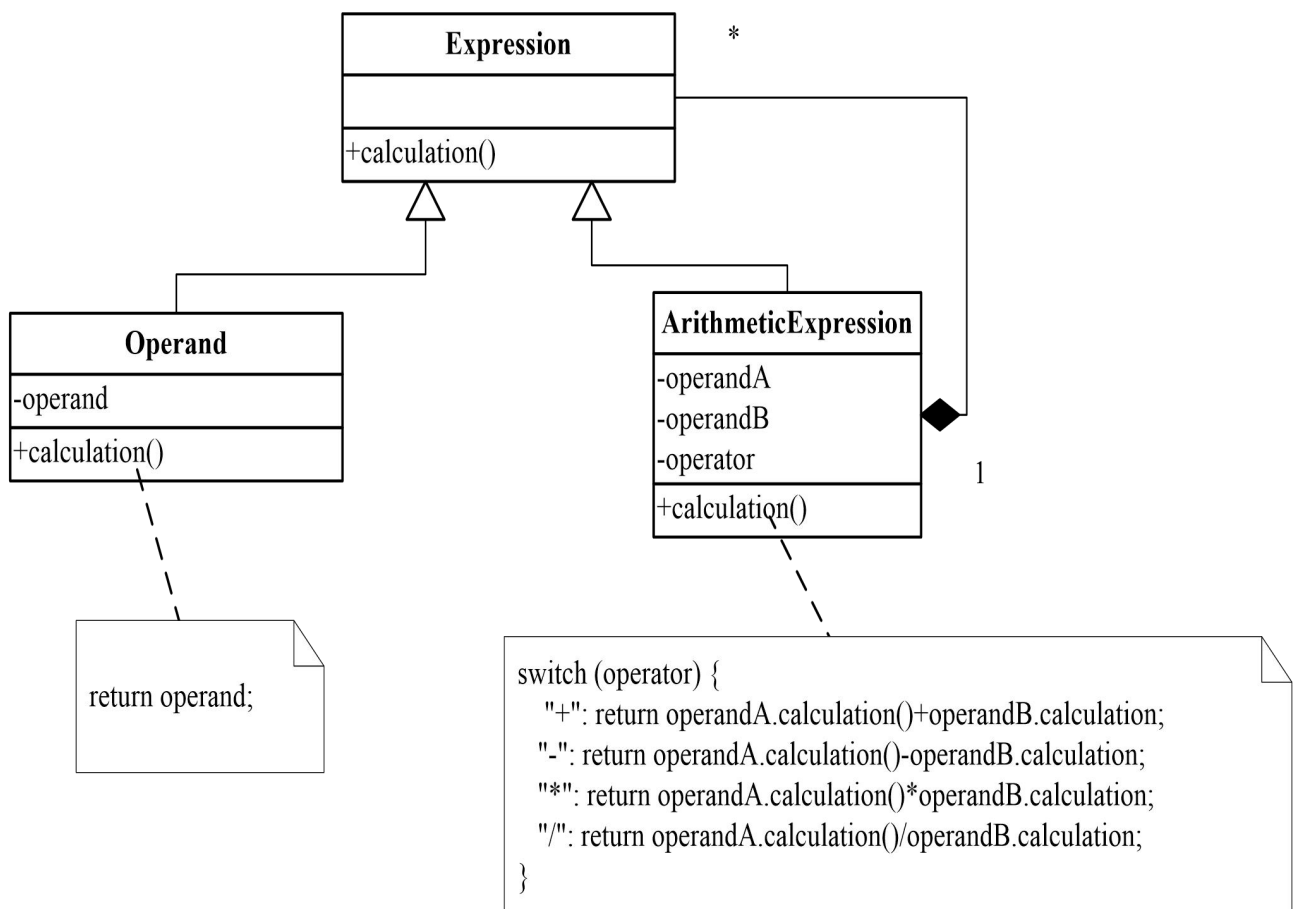
3.



4.



5.



6.

(1) 选择测试路径:

I: $a \rightarrow b \rightarrow d$ (判定 $(A > 1) \vee (B = 0)$ 为假; 判定 $(A = 2) \wedge (X > 1)$ 为假)

II: $a \rightarrow c \rightarrow e$ (判定 $(A > 1) \vee (B = 0)$ 为真; 判定 $(A = 2) \wedge (X > 1)$ 为真)

(2) 选择测试用例

I: 输入 $A=1$ $B=1$ $X=1$ 预期输出 $X=1$

II: 输入 $A=2$ $B=0$ $X=4$ 预期输出 $X=4$