

第 1 章 软件与软件工程

1. 下列说法正确的是 ()	20 世纪 60 年代提出了软件工程的观念。
2. 软件危机的主要原因是 ()	软件本身的特点及开发方法
3. 软件工程的 3 个要素是 ()	方法、工具和过程
4. 在下列选项中, () 不属于软件工程学科研究的基本内容。	软件工程材料

第 2 章 软件过程

5. 增量模型本质上是一种 ()	非整体开发模型
6. 软件过程是 ()	软件开发活动的集合
7. 软件生命周期模型不包括 ()	用例模型
8. 包含风险分析的软件工程模型是 ()	螺旋模型
9. 软件工程中描述生命周期的瀑布模型一般包括计划、需求分析、设计、编码、()、维护等几个阶段。	测试
10. 软件开发的瀑布模型, 一般都将开发过程划分为分析、设计、编码核测试阶段, 一般认为可能占用人员最多的阶段是 ()	编码阶段
11. 螺旋模型综合了 () 的优点, 并增加了风险分析	瀑布模型和快速原型模型

第 3 章 可行性研究与项目开发计划

12. 可行性研究也称为 ()	项目论证
13. () 研究往往是系统开发过程中难度最大的工作, 是可行性研究的关键	技术可行性
14. 研究软硬件资源的有效性是进行 () 研究的一方面	技术可行性
15. 软件开发计划是软件工程中的一种 () 性文档	管理

第 4 章 结构化分析

16. 在需求分析之前有必要进行（）工作	可行性研究
17. 需求分析是一个（），它应该贯穿于系统的整个生命周期，而不是仅仅属于软件生命周期早期的一项工作	过程
18. 软件需求规格说明书的内容不应该包括（）	对算法的详细过程描述
19. 软件需求分析阶段的工作，可以分为以下 4 个方面：对问题的识别、分析与综合、编写需求分析文档以及（）	需求分析评审
20. 进行需求分析可使用多种工具，但（）是不适用的	PAD 图
21. 数据流图是进行软件需求分析的常用图形工具，其基本图形符号是（）	加工、数据流、数据存储和外部实体
22. 结构化分析方法的基本思想是（）	自顶向下逐步分解
23. 在 E-R 图中，包含以下基本成分（）	实体、属性、关系

第 5 章 结构化设计

24. 面向数据流的软件设计方法可将（）映射成软件结构	数据流
25. 模块的独立性是由内聚性和耦合性来度量的，其中内聚性是（）	模块的功能强度
26. Jackson 方法根据（）来导出程序结构	数据结构
27. 为了提高模块的独立性，模块之间最好是（）	数据耦合
28. 再面向数据流的软件设计方法中，一般将信息流分为（）	变换流和事务流

第 6 章 面向对象方法与 UML

29. 在面向对象技术中，对象是类的实例。对象有 3 种成分：（）、属性和方法（或操作）	封装
30. 以下哪一项不是面向对象的特征？（）	过程调用
31. 汽车有一个发动机，汽车和发动机之间的关系是（）关系	整体部分
32. （）是把对象的属性和操作结合在一起，构成一个独立的对象，其内部信息对外界是隐藏的，外接只能通过有限的接	封装

口与对象发生联系	
33. 面向对象的主要特征除了对象唯一性、封装、继承外，还有（）	多态性
34. 关联是建立（）之间关系的一种手段	类
35. 面向对象软件技术的许多强有力的功能和突出的有点，都来源于把类组织成一个层次结构的系统，一个类的上层可以有父类，下层可以有子类，这种层次结构系统的一个重要性质是（），一个子类获得其弗雷的全部描述（数据和操作）	继承性
36. 所有对象可以成为各种对象类，每个对象类都定义了一组（）	方法
37. 通过执行对象的操作改变对象的属性，但它必须通过（）的传递	消息
38. UML 是软件开发中的一个重要工具，它主要应用于（）	基于对象的面向对象的方法
39. （）是从用户使用系统的角度描述系统功能的图形表达方法	用例图
40. （）描述了一组交互对象间的动态协作关系，它表示完成某项行为的对象和这些对象之间传递消息的时间顺序	顺序图

第 7 章 面向对象分析

41. 面向对象模型主要有以下哪些模型组成？（）	对象模型、动态模型、功能模型
42. 面向对象分析的首要工作是建立（）	问题的对象模型
43. 面向对象的分析方法主要是建立 3 类模型，即（）	对象模型、动态模型、功能模型
44. 在面向对象分析阶段建立的 3 个模型中，核心的模型是（）模型	对象
45. 在面向对象的动态模型中，每张状态图表示（）的动态行为	某一个类
46. 在考察系统一些设计是程序和改变的状况是，要用动态模型来表示。动态模型着重于系统的控制逻辑，它包括两个图：事件追踪图和（）	状态图
47. 对象模型的描述工具是（）	对象图
48. 功能模型中所有的（）往往形成一个层次结构，在这个层次结构中，一个数据流图的过程可以由下一层数据流图做进一步说明	数据流图

第 8 章 软件体系结构与设计模式

49. 程序编译器的体系结构适合使用（）	仓库体系结构
50. 网站系统是一个典型的（）	瘦客户机/服务器结构
51. 在分层体系结构中，（）实现与实体对象相关的业务逻辑	控制层
52. （）可以帮助人们简单方便的复用已经成功的设计或体系结构设计	设计模式
53. （）用于封装具体的平台，从而使应用程序可以在不同的平台上运行	抽象工厂模式

第 9 章 面向对象设计

54. 面向对象设计阶段的主要任务是系统设计和（）	对象设计
55. 只有类的共有界面的成员才能够成为使用类的操作，这是软件设计的（）原则	信息隐藏
56. （）是表达系统类及其相互联系的图示，它是面向对象设计的核心，是建立状态图、协作图和其他图的基础	类图
57. 下面所列的性质中，（）不属于面向对象设计的特性	可视化
58. 下列是面向对象设计方法中有关对象的叙述，其中（）是正确的	对象的属性集合是它的特征表示
59. 在面向对象设计中，基于父类创建的子类具有父类的所有特性（属性和方法），这一特点称为类的（）	继承性

第 10 章 软件实现

60. 软件实现是软件产品由概念到实体的关键过程，它将（）的结果翻译成用某种程序设计语言编写的并且最终可以运行的程序代码。虽然软件的质量取决于软件设计，但是规范的程序设计风格将会给后期的软件维护带来不可忽视的影响	详细设计
61. （）是一种纯面向对象语言	Eiffel
62. 第一个体现结构化编程思想的程序设计语言是（）	Pascal
63. 面向对象设计的结果，实现时（）	只能使用面向对象语言

第 11 章 软件测试

64. 软件测试的目的是（）	发现软件的错误
65. 白盒法又称为逻辑覆盖法，主要用于（）	单元测试
66. 在软件工程中，白盒测试方法可用于测试程序的内部结构，此方法将程序作为（）	路径的集合
67. 成功的测试是指运行测试用例后（）	发现了程序错误
68. 白盒测试法是根据程序的（）来设计测试用例的方法	内部逻辑
69. 在软件测试中，逻辑覆盖法主要用于（）	白盒测试方法
70. 黑盒测试是从（）观点的测试，白盒测试是从（）观点的测试	用户、开发人员
71. 软件测试可能发现软件中的（），但不能证明软件（）	错误、没有错误
72. 使用白盒测试方法时确定测试数据应根据（）和指定的覆盖标准	程序的内部逻辑
73. 黑盒测试方法根据（）设计测试用例	软件要完成的功能
74. 集成测试主要由两个方法：（）和（）	渐增式测试方法、非渐增式测试方法
75. 软件测试的目的是尽可能发现软件中的错误，通常（）是代码编写阶段可进行的测试，它是整个测试工作的基础	单元测试
76. 单元测试主要针对模块的几个基本特征进行测试，改阶段不能完成的测试是（）	系统功能