

软件工程概述

一 单项选择

1. 软件生命周期一般包括：软件开发期和软件运行期，下述（ D ）不是软件开发期所应包含的内容。
A 需求分析 B 结构设计 C 程序编制 D 软件维护
2. 软件是一种逻辑产品，它的开发主要是（ A ）。
A 研制 B 拷贝 C 再生产 D 复制
3. 以文档作为驱动，适合于软件需求很明确的软件项目的生存周期模型是（ C ）。
A 喷泉模型 B 增量模型 C 瀑布模型 D 螺旋模型
4. 在软件生存周期中，（ B ）阶段必须要回答的问题是“要解决的问题是做什么？”
A 详细设计 B 可行性分析和项目开发计划 C 概要设计 D 软件测试
5. 软件产品与物质产品有很大区别，软件产品是一种（ C ）产品
A 有形 B 消耗 C 逻辑 D 文档
6. （ C ）把瀑布模型和专家系统结合在一起，在开发的各个阶段上都利用相应的专家系统来帮助软件人员完成开发工作。
A 原型模型 B 螺旋模型 C 基于知识的智能模型 D 喷泉模型
7. （ B ）阶段是为每个模块完成的功能进行具体的描述，要把功能描述转变为精确的、结构化的过程描述。
A 概要设计 B 详细设计 C 编码 D 测试
8. 下列软件开发模型中，适合于那些不能预先确切定义需求的软件系统的开发的模型是（ A ）。
A 原型模型 B 瀑布模型 C 基于知识的智能模型 D 变换模型
9. 下列软件开发模型中，以面向对象的软件开发方法为基础，以用户的需求为动力，以对象来驱动的模式是（ C ）。
A 原型模型 B 瀑布模型 C 喷泉模型 D 螺旋模型
10. 下列软件开发模型中，支持需求不明确，特别是大型软件系统的开发，并支持多种软件开发方法的模型是（ D ）。
A 原型模型 B 瀑布模型 C 喷泉模型 D 螺旋模型
11. 软件特性中，使软件在不同的系统约束条件下，使用户需求得到满足的难易程度称为（ C ）。
A 可修改性 B 可靠性 C 可适应性 D 可重用性
12. 软件特性中，一个软件能再次用于其他相关应用的程度称为（ B ）。
A 可移植性 B 可重用性 C 容错性 D 可适应性
13. 软件特性中，（ A ）是指系统具有清晰的结构，能直接反映问题的需求的程度。
A 可理解性 B 可靠性 C 可适应性 D 可重用性
14. 软件特性中，软件产品交付使用后，在实现改正潜伏的错误、改进性能、适应环境变化等方面工作的难易程度称为（ B ）。
A 可理解性 B 可维护性 C 可适应性 D 可重用性
15. 软件特性中，软件从一个计算机系统或环境移植到另一个上去的难易程度指的是（ C ）。
A 可理解性 B 可修改性 C 可移植性 D 可重用性
16. 软件特性中，在给定的时间间隔内，程序成功运行的概率指的是（ D ）。
A 有效性 B 可适应性 C 正确性 D 可靠性
17. 软件特性中，允许对软件进行修改而不增加其复杂性指的是（ A ）。
A 可修改性 B 可适应性 C 可维护性 D 可移植性
18. 软件特性中，多个软件元素相互通讯并协同完成任务的能力指的是（ B ）。
A 可理解性 B 可互操作性 C 可维护性 D 可追踪性
19. 软件特性中，根据软件需求对软件设计、程序进行正向追踪，或根据程序、软件设计对软件需求进行逆向追踪的能力指的是（ C ）。

- A 可理解性 B 可互操作性 C 可追踪性 D 可维护性
20. 软件的可修改性支持软件的 (D)。
- A 有效性 B 可互操作性 C 可追踪性 D 可维护性
21. 软件的可移植性支持软件的 (A)。
- A 可适应性 B 可互操作性 C 可追踪性 D 有效性
22. 软件的可理解性支持软件的 (B)。
- A 有效性 B 可移植性 C 可追踪性 D 可靠性
23. 在软件工程的原则中, 抽象、信息隐藏、模块化和局部化的原则支持软件的 (C)。
- A 有效性 B 可互操作性 C 可靠性 D 可追踪性
24. 下列选项中, 属于需求分析阶段的任务的是 (D)。
- A 组装测试计划 B 单元测试计划 C 软件总体设计 D 验收测试计划
25. 下列选项中, 属于概要设计阶段的任务的是 (A)。
- A 组装测试计划 B 单元测试计划 C 初步用户手册 D 验收测试计划
26. 下列选项中, 属于详细设计阶段的任务的是 (B)。
- A 组装测试计划 B 单元测试计划 C 初步用户手册 D 验收测试计划
27. 下列选项中, 属于实现阶段的任务的是 (C)。
- A 组装测试计划 B 绘制程序流程图 C 单元测试 D 验收测试计划
28. 下列选项中, 在验收测试结束时, 必须提交给用户的是 (D)。
- A 项目开发总结报告 B 验收测试计划 C 需求规格说明书 D 最终用户手册

二 填空

1. 计算机软件是指与计算机系统操作有关的 _____ 以及任何与之有关的文档资料。
答案: 程序、数据
2. _____ 是将软件生存周期各个活动规定为依线性顺序连接的若干阶段的模型。
答案: 瀑布模型
3. _____ 是软件生存周期中时间最长的阶段。
答案: 维护阶段
4. _____ 是一种以用户需求为动力, 以对象作为驱动力的模型, 适合于面向对象的开发方法。
答案: 喷泉模型
5. 软件生存周期是指一个软件从 _____ 开始直到该软件最终退役为止的整个时期。
答案: 定义
6. 在软件的生产过程中, 总是有大量各种信息要记录, 因此, _____ 在产品的开发过程中起着重要的作用。
答案: 文档
7. 在软件生存周期中, 软件定义过程的基本任务是确定软件系统的 _____。
答案: 工程需求
8. 在软件生存周期中, 软件定义过程可以通过软件系统的 _____ 两个阶段来完成。
答案: 可行性研究和需求分析
9. 在软件生存周期中, 软件开发过程可以通过 _____、实现、组装测试、验收测试共 5 个阶段组成。
答案: 概要设计、详细设计
10. 我们把支持软件项目的 _____ 的软件叫做软件开发的工具软件。
答案: 开发、管理、维护活动
11. 软件工程由 _____ 三部分组成, 称为软件工程的三要素。

答案：方法、过程和工具

三 名词解释

1．计算机软件

答：计算机软件是指与计算机系统操作有关的程序、数据以及任何与之有关的文档资料。

2．软件危机

答：软件危机是指在计算机软件开发、使用与维护过程中遇到的一系列严重问题和难题。

3．软件工程

答：软件工程是指用工程、科学和数学的原则与方法开发、维护计算机软件的有关技术和管理方法

4．软件工程过程

答：软件工程过程指在定义、开发、使用和维护软件时，要实施的过程、活动和任务。

5．软件生存周期

答：软件从定义开始，经过开发、使用和维护，直到最终退役的全过程称为软件生存周期。

6．软件生存周期模型

答：软件开发模型（又称为软件生存周期模型）是指软件项目开发和维护的总体过程思路的框架。

7．计算机辅助软件工程

答：在软件工程活动中，人们按照软件工程的原则和方法，利用计算机及其集成的软件开发环境，辅助软件项目的开发、维护及管理的过程，称为计算机辅助软件工程（即 CASE）。

四 简答题

1．软件产品的特点是什么

答： 软件开发与硬件研制相比，更依赖于开发人员的业务素质、智力、人员的组织、合作和管理。
大型软件仍然存在着潜伏的错误。
软件产品开发成功后，只需对原版软件进行复制，即可生产出任意多的同样的产品。
还有，软件在使用过程中的维护工作比硬件要复杂得多。
由于软件不是物理产品，所以它不会磨损和老化。

2．软件危机的表现有哪些？

答：1) 对软件开发成本和进度的估计常常很不准确。
2) 用户对已完成的软件不满意的现象时有发生。
3) 软件产品的质量往往是靠不住的。
4) 软件常常是不可维护的。
5) 软件通常没有适当的文档资料。
6) 软件成本在计算机系统总成本中所占比例逐年上升。
7) 软件开发生产率提高的速度远跟不上日益增长的软件需求。

3．产生软件危机的原因主要有哪些？

答：1) 用户对软件需求的描述不精确。
2) 软件开发人员对用户需求的理解有偏差。
3) 缺乏处理大型软件项目的经验。
4) 开发大型软件易产生疏漏和错误。
5) 缺乏有力的方法学的指导和有效的开发工具的支持。
6) 面对日益增长的软件需求，人们显得力不从心。

4．软件工程的 7 条基本原理是什么？

答：1) 用分阶段的生存周期计划严格管理

- 2) 坚持进行阶段评审
- 3) 实行严格的产品控制
- 4) 采用现代程序设计技术
- 5) 结果应能清楚地审查
- 6) 开发小组的人员应少而精
- 7) 承认不断改进软件工程实践的必要性

5. 软件工程的 7 条原则是什么？

答：1. 抽象。

2. 信息隐藏。
3. 模块化。
4. 局部化。
5. 一致性。
6. 完全性。
7. 可验证性。

6. 软件生存周期如何划分？

答：可将软件生存周期划分为 3 个过程共 9 个阶段。

软件定义过程：由可行性研究和需求分析 2 个阶段组成；

软件开发过程：由概要设计、详细设计、实现、组装测试、验收测试 5 个阶段组成；

软件使用与维护过程：由使用与维护 and 退役 2 个阶段组成。

7. 常见的软件开发模型有哪些？

答：有瀑布模型、原型模型、螺旋模型、喷泉模型、变换模型、基于四代技术的模型、基于知识的智能模型等。

8. 瀑布模型适合于哪类软件系统的开发？

答：瀑布模型一般适用于功能、性能明确、完整、无重大变化的软件系统的开发。例如操作系统、编译系统、数据库管理系统等系统软件的开发。

9. 瀑布模型的缺点主要有哪些？

答：1) 在软件开发的初期阶段就要求做出正确、全面、完整的需求分析对许多应用软件来说是极其困难的。

2) 在需求分析阶段，当需求确定后，无法及时验证需求是否正确、完整。

3) 不支持产品的演化，缺乏灵活性，使软件产品难以维护。

10. 实现快速原型的开发途径有哪三种？

1) 仅模拟软件系统的人机界面和人机交互方式。

2) 开发一个工作模型，实现软件系统中重要的或容易产生误解的功能。

3) 利用一个或几个类似的正在运行的软件向用户展示软件需求中的部分或全部功能。

11. 原型模型适于哪种情况的软件开发？

答：原型模型适合于那些不能预先确切定义需求的软件系统的开发，更适用于那些项目组成员（包括分析员、设计员、程序员和用户）不能很好交流或通信有困难的情况。

12. 螺旋模型的每一个周期包括哪几个阶段？

答：螺旋模型的每一个周期都包括计划（需求定义）、风险分析、工程实现和用户评价与阶段评审 4 个阶段。

13. 螺旋模型的优点有哪些？

答：螺旋模型的主要优点有：

1) 支持用户需求的动态变化。

2) 原型可看作形式的可执行的需求规格说明，易于为用户和开发人员共同理解，还可作为继续开发的基础，并为用户参与所有关键决策提供了方便。

3) 螺旋模型特别强调原型的可扩充性和可修改性, 原型的进化贯穿整个软件生存周期, 这将有助于目标软件的适应能力。

4) 为项目管理人员及时调整管理决策提供了方便, 进而可降低开发风险。

14. 螺旋模型适于哪种情况的软件开发?

答: 螺旋模型支持需求不明确、特别是大型软件系统的开发, 并支持面向规格说明、面向过程、面向对象等多种软件开发方法, 是一种具有广阔前景的模型。

软件工程单元二 (软件项目管理)

一 单项选择

1. 软件项目管理必须 (A) 介入。

A) 从项目的开头 B) 在可行性研究之后 C) 在需求分析之后 D) 在编码之后

2. 下列选项中, 属于成本管理的主要任务的是 (B)。

A) 人员的分工 B) 估算软件项目的成本
C) 对控制软件质量要素 D) 对软件阶段产品进行评审

3. 下列选项中, 不属于质量管理的主要任务的是 (C)。

A) 制定软件质量保证计划 B) 按照质量评价体系控制软件质量要素
C) 增加软件产品的功能 D) 对最终软件产品进行确认

4. 下列选项中, 属于软件配置管理的任务的是 (D)。

A) 人员的分工 B) 估算软件项目的成本
C) 对软件阶段产品进行评审 D) 对程序、数据、文档的各种版本进行管理

5. (A) 是对产品或过程的某个属性的范围、数量、维度、容量或大小提供一个定量的指示。

A) 测量 B) 度量 C) 估算 D) 指标

6. (B) 是对系统、部件或过程的某一特性所具有的程度进行的量化测量。

A) 测量 B) 度量 C) 估算 D) 指标

7. (C) 是对软件产品、过程、资源等使用历史资料或经验公式等进行预测。

A) 测量 B) 度量 C) 估算 D) 指标

8. (D) 是一个度量或度量的组合, 它可对软件产品、过程或资源提供更深入的理解。

A) 测量 B) 度量 C) 估算 D) 指标

9. 下列选项中属于产品的内部属性的是 (A)。

A) 模块耦合度 B) 软件可靠性 C) 软件有效性 D) 软件可维护性

10. 下列选项中属于过程的外部属性的是 (B)。

A) 工作量 B) 成本 C) 事件 D) 计划及进度

11. 下列选项中属于资源的外部属性的是 (C)。

A) 经验 B) 人员 C) 生产率 D) 工具

12. 基于代码行的面向规模的度量方法适合于 (D)。

A) 过程式程序设计语言和事前度量 B) 第四代语言和事前度量
C) 第四代语言和事后度量 D) 过程式程序设计语言和事后度量

13. 下列说法中, 不正确的是 (A)。

A) 功能点度量方法与程序设计语言有关 B) 功能点度量方法适合于过程式语言 C) 功能点度量方法适合于非过程式语言
D) 功能点度量方法适合于软件项目估算

14. 下列说法中, 不正确的是 (B)。

A) 软件项目工作量分布曲线不是线性的
B) 到开发后期临时增加人力可加快进度
C) 适当推迟交货时间可大幅度降低开发工作量

- D) 在软件生存周期的不同阶段各类人员的参与情况是不同的。
15. 在软件质量要素中，程序满足需求规格说明及用户目标的程度指的是 (C)。
- A) 完整性 B) 可用性 C) 正确性 D) 灵活性
16. 在软件质量要素中，改变一个操作的顺序所需的工作量的多少指的是 (D)。
- A) 完整性 B) 可用性 C) 正确性 D) 灵活性
17. 在软件质量要素中，对未授权人员访问程序或数据加以控制的程度指的是 (A)。
- A) 完整性 B) 可用性 C) 正确性 D) 灵活性
18. 在软件质量要素中，学习使用软件（即操作软件、准备输入数据、解释输出结果等）的难易程度指的是 (B)。
- A) 完整性 B) 可用性 C) 正确性 D) 灵活性
19. 在软件质量要素的评价准则中，程序源代码的紧凑程度指的是 (C)。
- A) 简单性 B) 准确性 C) 简明性 D) 执行效率
20. 在软件质量要素的评价准则中，软件运行的效率指的是 (D)。
- A) 简单性 B) 准确性 C) 简明性 D) 执行效率
21. 在软件质量要素的评价准则中，程序易于理解的程度指的是 (A)。
- A) 简单性 B) 准确性 C) 简明性 D) 执行效率
22. 在软件质量要素的评价准则中，计算和控制的精确程度指的是 (B)。
- A) 简单性 B) 准确性 C) 简明性 D) 执行效率
23. 在软件质量要素的评价准则中，在各种异常情况下软件能继续提供操作的能力指的是 (A)。
- A) 容错性 B) 检测性 C) 安全性 D) 可操作性
24. 在软件质量要素的评价准则中，程序监视自身运行并标识错误的程度指的是 (B)。
- A) 容错性 B) 检测性 C) 安全性 D) 可操作性
25. 在软件质量要素的评价准则中，控制或保护程序和数据不被破坏、非法访问等机制的能力指的是 (C)。
- A) 容错性 B) 检测性 C) 安全性 D) 可操作性
26. 在软件质量要素的评价准则中，操作该软件的难易程度指的是 (D)。
- A) 容错性 B) 检测性 C) 安全性 D) 可操作性
27. 下列选项中，不能进行并行工作的是 (A)。
- A) 需求复审 B) 过程设计 C) 单元测试 D) 编码
28. 在整个软件项目定义与开发各阶段一种典型的工作量分布原则，称为 40-20-40 分布原则，其中 20% 的工作量指的是 (B)。
- A) 需求分析工作量 B) 编码工作量 C) 详细设计工作量 D) 测试工作量

二 填空

1. 软件项目管理的对象主要包括 _____。
- 答案：产品、过程和资源等
2. 对不依赖于其他属性的软件的简单属性的测量称为 _____。
- 答案：直接度量
3. 对涉及若干个其他属性的软件要素、准则或属性的度量，称为 _____。
- 答案：间接度量
4. 面向规模的度量是以软件的 _____ 为基础的直接度量。
- 答案：代码行数
5. Albrecht 首先提出了功能点度量方法。这是一种面向功能的 _____ 方法。
- 答案：间接度量

6. 在软件项目管理过程中的第一个活动应当是_____。

答案：估算

7. Putnam 估算模型公式是：_____。

答案： $L = C_k E^{1/3} t_d^{4/3}$

8. 估算工作量的公式： $E = L^3 / (C_k^3 t_d^4)$ ，式中的 t_d 是对应于_____时的时间，它正好是工作量曲线的峰值，说明此时的工作量最大、参加项目的人最多。

答案：软件交付

9. 软件开发项目每年所需的人年数与开发时间的关系满足_____ Rayleigh-Norden 分布，即软件项目的工作量分布曲线_____的，因此，参加软件项目的人员就不能一成不变。

答案：不是线性

10. 开发软件项目的工作量和交货时间 t_d 的 4 次方成反比，如果条件允许，适当地推迟交货时间（即使 t_d 增大），可大幅度降低_____。

答案：开发工作量

11. Putnam 模型的优点是揭示了软件项目的_____、软件开发时间和工作量三者之间的关系，在理论上具有重要意义。

答案：源程序代码长度

12. CoCoMo 模型按其详细程度分为基本 CoCoMo 模型、_____和详细 CoCoMo 模型三个层次。

答案：中间 CoCoMo 模型

13. 基本 CoComo 模型主要用于_____估算整个系统开发和维护的工作量以及软件开发所需要的时间。

答案：系统开发的初期

14. 一般地，由 N 个程序员组成的程序员小组在完成相同规模的软件时需要的通信数量 $C(N)$ =_____。

答案： $N(N-1)/2$

15. McCall 等人提出了由_____、评价准则、定量度量三个层次组成的三层次度量模型。

答案：软件质量要素

16. 程序图 G 的环形复杂度 $V(G)$ 的值等于程序图中有界和无界的_____的个数。

答案：封闭区域

17. McCabe 指出， $V(G)$ 可作为程序规模的定量指标， $V(G)$ 值越高的程序往往是越复杂、越容易出问题的程序。因此，他建议模块规模以_____为宜。

答案： $V(G) \leq 10$

18. Halstead 给出了称为文本复杂性度量的模型。它是根据统计程序中的_____的个数来度量程序的复杂程度。

答案：操作符和操作数

19. 软件可靠性定义为在某个给定_____内，程序按照规格说明成功运行的概率。

答案：时间间隔

20. 软件的有效性函数 $A(t)$ 定义为软件系统在_____按照规格说明成功运行的概率。

答案：时刻 t

21. 软件项目风险的一个显著特点是具有_____，某项风险可能发生也可能不发生。

答案：不确定性

22. 风险分析的四个主要活动：_____、风险估算、风险评价、风险驾驭和监控。

答案：风险标识

23. 风险分析的四个主要活动：风险标识、_____、风险评价、风险驾驭和监控。

答案：风险估算

24．风险分析的四个主要活动：风险标识、风险估算、_____、风险驾驭和监控。

答案：风险评价

25．风险分析的四个主要活动：风险标识、风险估算、风险评价、_____。

答案： 风险驾驭和监控

26．软件项目的风险按照其影响的范围，可分为_____、技术风险和商业风险三类。

答案：项目风险

27．软件项目的风险按照其影响的范围，可分为项目风险、_____和商业风险三类。

答案：技术风险

28．软件项目的风险按照其影响的范围，可分为项目风险、技术风险和_____三类。

答案：商业风险

29．为了识别风险， Boehm 建议使用各类_____来标识风险。

答案：风险检测表

30．为了评价风险，常采用三元组 $[r_i, p_i, x_i]$ 来描述风险。其中 r_i 代表_____， p_i 表示第 i 种风险发生的概率， x_i 代表该风险带来的影响。

答案：第 i 种风险

31．为了评价风险，常采用三元组 $[r_i, p_i, x_i]$ 来描述风险。其中 r_i 代表第 i 种风险， p_i 表示_____， x_i 代表该风险带来的影响。

答案：第 i 种风险发生的概率

32．为了评价风险，常采用三元组 $[r_i, p_i, x_i]$ 来描述风险。其中 r_i 代表第 i 种风险， p_i 表示第 i 种风险发生的概率， x_i 代表_____。

答案：该风险带来的影响

33．一个对风险评价很有用的技术就是定义风险参照水准。对于大多数软件项目来说，
_____就是典型的风险参照水准。

答案：成本、进度、性能

34．风险分析的目的是建立处理风险的策略，_____风险。

答案：监控、驾驭

35．在整个软件项目定义与开发各阶段一种典型的工作量分布原则，称为_____分布原则。

答案：40-20-40

36．目前，软件项目的进度安排的两种比较常用的方法是_____和关键路径法（CPM），这两种方法都生成描述项目进展状态的任务网络图。

答案：程序评估与审查技术（PERT）

37．目前，软件项目的进度安排的两种比较常用的方法是程序评估与审查技术（PERT）和_____，这两种方法都生成描述项目进展状态的任务网络图。

答案：关键路径法（CPM）

38．某个子任务的最早启动时间是指该子任务的_____完成的最早时间。

答案：所有各前导子任务

39．某子任务的最早启动时间与完成该子任务所需时间之和就是该子任务的_____。

答案：最早结束时间

40．某个子任务的_____是指在保证项目按时完成的前提下最晚启动该子任务的时间。

答案：最迟启动时间

41．最迟启动时间与完成该子任务所需时间之和就是该子任务的_____。

答案：最迟结束时间

42．在制定进度计划时，应首先找到影响进度的_____，并在其上安排一定的节假日和机动时间，以

便应付可能出现的问题和难点。

答案：关键路径

43．正式的技术评审是降低 _____ 的重要措施。

答案：软件成本

44．实践表明，向一个已经延期的软件项目追加新的开发人员，可能使项目完成得 _____。

答案：更晚

三 名词解释

1．测量 (measure)

答：是对产品或过程的某个属性的范围、数量、维度、容量或大小提供一个定量的指示。

2．度量 (metric)

答：是对系统、部件或过程的某一特性所具有的程度进行的量化测量。

3．估算 (estimation)

答：是对软件产品、过程、资源等使用历史资料或经验公式等进行预测。

4．指标 (guideline)

答：是一个度量或度量的组合，它可对软件产品、过程或资源提供更深入的理解。

5．软件可靠性

答：软件可靠性定义为在某个给定时间间隔内，程序按照规格说明成功运行的概率。

6．软件的有效性函数 $A(t)$

答：软件的有效性函数 $A(t)$ 定义为软件系统在时刻 t 按照规格说明成功运行的概率。

四 简答题

1．软件项目管理的主要任务是什么？

答：软件项目管理的主要任务是根据选定的软件开发过程框架（即软件开发模型）和对其估算的结果制定软件项目实施计划；再根据计划对人员进行组织、分工；按照计划的进度，以及成本管理、风险管理、质量管理的要求，控制并管理软件开发和维护的活动，最终以最小的代价完成软件项目规定的全部任务。

2．成本管理的主要任务是什么？

答：成本管理的主要任务是估算软件项目的成本，作为立项和签合同的依据之一，并在软件开发过程中按计划管理经费的使用。

3．质量管理的主要任务是什么？

答：质量管理的主要任务是制定软件质量保证计划，按照质量评价体系控制软件质量要素，对阶段性的软件产品进行评审，对最终软件产品进行确认，确保软件质量。

4．常用的软件项目的估算方法主要有哪 4 种？

答：1) 自顶向下的估算方法

2) 自底向上的估算方法

3) 差别估算法

4) 根据实验或历史数据给出软件项目工作量或成本的经验估算公式。

5．CoCoMo 模型按其详细程度分为哪三个层次？

答：CoCoMo 模型按其详细程度分为基本 CoCoMo 模型、中间 CoCoMo 模型和详细 CoCoMo 模型三个层次。

6．软件质量的定义？

答：1983 年，ANSI/IEEE std729 标准给出了软件质量的定义如下：

软件质量是软件产品满足规定的和隐含的与需求能力有关的全部特征和特性，包括：

1) 软件产品满足用户要求的程度；

2) 软件拥有所期望的各种属性的组合程度；

3) 用户对软件产品的综合反映程度；

4) 软件在使用过程中满足用户需求的程度。

7. 简述 McCall 等人的软件质量度量模型

答：McCall 等人提出了由软件质量要素、评价准则、定量度量三个层次组成的三层次度量模型。其中第一层是将对软件质量的度量归结为对直接影响软件质量的若干个软件质量要素的度量；由于质量要素很难直接度量，所以第二层是用若干个可度量的评价准则来间接度量软件质量要素；而第三层是对相应评价准则的直接度量。

8. 软件复杂性的概念？

答：K. Magel 从 6 个方面来描述软件复杂性：

- 1) 理解程序的难度；
- 2) 维护程序的难度；
- 3) 向其他人解释程序的难度；
- 4) 按指定方法修改程序的难度；
- 5) 根据设计文件编写程序的工作量；
- 6) 执行程序时需要资源的多少。

9. 衡量软件的复杂程度有哪些基本原则？

答：衡量软件的复杂程度有 12 条基本原则，如下：

- 1) 软件的复杂性与其规模的关系不是线性的；
- 2) 数据结构复杂的程序较复杂；
- 3) 控制结构复杂的程序较复杂；
- 4) 转向语句使用不当的程序较复杂；
- 5) 循环结构比选择结构复杂、选择结构比顺序结构复杂；
- 6) 语句、数据、子程序模块等出现的顺序对复杂性有影响；
- 7) 非局部变量较多的程序较复杂；
- 8) 参数按地址调用 (Call by reference) 比按值调用 (Call by value) 复杂；
- 9) 函数副作用比显式参数传递难理解；
- 10) 作用不同的变量同名时较难理解；
- 11) 模块、过程间联系密切的程序较复杂；
- 12) 程序嵌套层数越多越复杂。

10. 软件开发项目管理过程主要包括哪几个方面？

答：软件开发项目管理过程主要包括以下几个方面：

- 1) 启动一个软件项目。
- 2) 成本估算。
- 3) 风险分析。
- 4) 进度安排。
- 5) 追踪和控制。

11. 风险分析有哪四个主要活动？

答：风险分析的四个主要活动：风险标识、风险估算、风险评价、风险驾驭和监控。

12. 解释评价风险的三元组 $[r_i, p_i, x_i]$ 。

答：在评价风险的三元组 $[r_i, p_i, x_i]$ 中， r_i 代表第 i 种风险， p_i 表示第 i 种风险发生的概率， x_i 代表该风险带来的影响， $i=1, 2, \dots, l$ ，表示软件开发项目共有 l 种风险， i 为风险序号。

13. 风险评价过程可分为哪四步？

答：风险评价过程可分四步进行：

- 1) 定义项目的风险参照水准；
- 2) 定义每种风险的三元组 $[r_i, p_i, x_i]$ ，并找出和每个参照水准之间的关系；

- 3) 预测一组参照点以定义一个项目终止区域，用一条曲线或一些易变动区域来定界；
- 4) 预测各种风险组合的影响是否超出参照水准。

14. 风险驾驭与监控的主要目标有哪些？

答：风险驾驭与监控的主要目标有三个：

- 1) 判断一个预测的风险是否已经发生；
- 2) 确保针对每一个风险而制定的风险驾驭步骤正在合理地实施；
- 3) 收集有关风险分析的所有信息，以备将来使用。

15. 软件项目开发的进度安排必须解决好哪几个问题？

答：软件项目开发的进度安排必须解决好以下几个问题：

- 1) 任务、人力、时间等资源的分配应与工程进度相一致；
- 2) 任务的分解与并行开发；
- 3) 工作量的分配；
- 4) 具体进度安排。

16. 软件质量保证（ SQA ）活动主要包括哪些内容？

答： SQA 活动主要包括以下内容：

1) 在需求分析阶段提出对软件质量的需求，并将其自顶向下逐步分解为可以度量和控制的质量要素，为软件开发、维护各阶段软件质量的定性分析和定量度量打下基础；

- 2) 研究并选用软件开发方法和工具；
- 3) 对软件生存周期各阶段进行正式的技术评审（ FTR ）；
- 4) 制定并实施软件测试策略和测试计划；
- 5) 及时生成软件文档并进行其版本控制；
- 6) 保证软件开发过程与选用的软件开发标准相一致；
- 7) 建立软件质量要素的度量机制；
- 8) 记录 SQA 的各项活动，并生成各种 SQA 报告。

17. 正式的技术评审（ FTR ）的过程一般由哪 6 个步骤组成？

答：FTR 的过程一般由 6 个步骤组成：

制定评审计划，即安排好评审会议日程。

介绍工程情况。

准备工作。评审小组成员自己审阅文档资料，并把发现的问题和错误记录下来，以备在评审会议上讨论。

正式召开评审会议。

工程返工。

工程复审。

如此反复、直至通过复审为止。

18. 在建立软件开发组织的时候要注意的原则是哪些？

答：在建立软件开发组织的时候要注意的原则是：

尽早落实责任。特别是软件项目负责人的责任；

减少接口。组织应该有良好的组织结构、合理的人员分工，以减少不必要的通信；

责权均衡。指软件经理的责任不应比赋予他的权力还大。

19. 程序设计小组的组织形式有哪几种？

答：程序设计小组的组织和小组内部人员的组织形式对生产率都会产生影响。常采用的组织形式有主程序员制小组、民主制小组、层次式小组 3 种。

20. 一个主程序员制小组由哪些人员组成？

答：由 1 位主程序员（高级工程师）、2~5 位程序员（技术员）、1 位后援工程师组成，还可以配备辅助人员（如资料员）。

21．民主制小组是怎样的一种组织形式？

答：民主制小组由若干程序员组成，虽然也设置一位组长，但是每当遇到问题时，组内的成员可以进行民主协商，以平等的地位交换意见。工作目标的制定、做出决定都有全体组员参加，即强调发挥小组每一个成员的积极、主动性和协作精神。

22．层次式小组是怎样的一种组织形式？

答：层次式小组是将组内人员分为 3 级：组长 1 人，他作为项目负责人负责全组工作；他直接领导 2 ~ 7 名高级程序员；每位高级程序员通过基层小组管理若干名程序员。

23．软件开发各个阶段都需要哪些人员？

答：在软件项目的计划和分析阶段，只需要少数人，主要是系统分析员、从事软件系统论证和概要设计的软件高级工程师和项目高级管理人员，人数虽不多，但都是高层次人员。概要设计阶段要增加几个高级程序员，详细设计阶段要增加软件工程师和程序员，在编码和测试阶段还要增加初级程序员和软件测试员。在这一过程中，各类专门人员和管理人员也在逐渐增加，到验收测试时，维护人员也加入其中，使各类人员的数量达到了最高峰。在软件产品交付使用的初期，参加软件维护的人员较多，此时为防止给维护活动带来困难，不应过早地解散软件开发人员。软件经过一段时间的纠错性维护后，出错率会明显减少，这时可以逐步撤出软件开发人员，之后，软件维护人员也逐步撤离。

24．在软件项目中配备和使用人员应遵循的 3 个主要原则是什么？

答：在软件项目中配备和使用人员时，应遵循的 3 个主要原则是：

重质量：使用少量有实践经验、素质高、有能力的人员去完成关键性任务，常常比使用较多的经验不足的人员更有效。

重培训：花力气培养所需的技术和管理人员是解决人员问题的有效方法。

双阶梯提升：人员要么按照技术职务提升，要么按照管理职务提升，两者不应兼得。

25．对软件项目经理人员的要求有哪些？

答：对项目经理除了要求具有较强的管理能力外，还要求其应具有的能力有：

把用户提出的非技术性要求加以整理提炼，以技术说明书形式转告给分析员和测试员。

能说服用户放弃那些不切实际的要求，以保证合理的要求得以满足。

具有综合问题的能力。

具有很强的沟通能力。

26．一个好的开发人员应具备的素质和能力有哪些？

答：一个好的开发人员应具备的素质和能力有：

善于与周围人员团结协作，建立良好的人际关系，善于听取别人的意见。

牢固掌握计算机软件的基本知识和技能。

善于分析和综合问题，具有严密的逻辑思维能力。

工作踏实、细致，遵循标准和规范，不靠碰运气，具有严格的科学作风。

工作中表现有责任心、有毅力、有耐心。

具有良好的书面和口头表达能力。

27．项目管理人员经常采用的跟踪方式主要有哪些？

答：项目管理人员经常采用的跟踪方式主要有：

定期召开项目工作会议，让每个项目成员汇报任务进展情况和存在的问题。

在软件开发过程中，请专家和用户按照里程碑对阶段性成果进行管理复审，判定实际开发进度是否与计划中定义的里程碑保持一致。

对照进度计划检查各子任务的实际开始时间是否与计划的开始时间一致。

及时了解项目开发人员的进展情况及存在的主要问题。

五、综合题

1. 已知有一个国外典型的软件项目的记录，开发人员 $M=6$ 人，其代码行数 $L=20.2\text{KLOC}$ ，工作量 $E=43\text{PM}$ ，成本 $S=314000$ 美元，错误数 $N=64$ ，文档页数 $P_d=1050$ 页。试计算开发该软件项目的生产率 P 、平均成本 C 、代码出错率 EQR 和文档率 D 。

解：根据给出的已知数据，可得：

$$P = L / E = 20.2 \text{ KLOC} / 43 \text{ PM} = 0.47 \text{ KLOC} / \text{PM} = 470 \text{ LOC} / \text{PM}$$

$$C = S / L = 314000 \text{ 美元} / 20.2 \text{ KLOC} = 15.54 \text{ 美元} / \text{LOC}$$

$$EQR = N / L = 64 \text{ 个} / 20.2 \text{ KLOC} = 3.17 \text{ 个} / \text{KLOC}$$

$$D = P_d / L = 1050 \text{ 页} / 20.2 \text{ KLOC} = 51.98 \text{ 页} / \text{KLOC}$$

2. 已知有一个软件项目的记录，开发人员 $M=3$ 人，其代码行数 $L=12.1\text{KLOC}$ ，工作量 $E=24\text{PM}$ ，成本 $S=168000$ 美元，错误数 $N=29$ ，文档页数 $P_d=365$ 页。试计算开发该软件项目的生产率 P 、平均成本 C 、代码出错率 EQR 和文档率 D 。

解：根据给出的已知数据，可得：

$$P = L / E = 12.1 \text{ KLOC} / 24 \text{ PM} = 0.504 \text{ KLOC} / \text{PM} = 504 \text{ LOC} / \text{PM}$$

$$C = S / L = 168000 \text{ 美元} / 12.1 \text{ KLOC} = 13.88 \text{ 美元} / \text{LOC}$$

$$EQR = N / L = 29 \text{ 个} / 12.1 \text{ KLOC} = 3.4 \text{ 个} / \text{KLOC}$$

$$D = P_d / L = 365 \text{ 页} / 12.1 \text{ KLOC} = 31.4 \text{ 页} / \text{KLOC}$$

3. 已知某软件项目的特征为：用户输入数为 30，用户输出数为 60，用户查询数为 24，共有 8 个文件，有 2 个外部界面。如果每个信息量的加权因子都取“一般”值，所有的技术复杂性调节因子都取“普通”值，用 Albrecht 方法计算该软件项目的功能点。

解：基本功能点 CT 为：

$$\begin{aligned} CT &= \text{用户输入数} \times 4 + \text{用户输出数} \times 5 \\ &\quad + \text{用户查询数} \times 4 + \text{文件数} \times 10 + \text{外部接口数} \times 7 \\ &= 30 \times 4 + 60 \times 5 + 24 \times 4 + 8 \times 10 + 2 \times 7 \\ &= 610 \end{aligned}$$

$$\begin{aligned} FP &= CT \times TCF = CT [0.65 + 0.01 \sum F_i] \\ &= 610 \times [0.65 + 0.01 \times 3 \times 14] \\ &= 652.7 \text{ 功能点} \end{aligned}$$

答：该软件项目共有 652.7 功能点。

4. 1978 年，Putnam 提出了大型软件项目的动态多变量估算模型 $L = C_k E^{1/3} t_d^{4/3}$ 。如果将开发时间 t_d 延长 20%，工作量可以减少到原来的百分之几？

解：由 $L = C_k E^{1/3} t_d^{4/3}$ 式可得：

$$E = L^3 / (C_k^3 t_d^4) \quad (1)$$

由已知，开发时间延长为 $t_{d1} = 1.2 t_d$ ，有：

$$E_1 = L^3 / (C_k^3 t_{d1}^4) \quad (2)$$

$$E_1 / E = [L^3 / (C_k^3 t_{d1}^4)] / [L^3 / (C_k^3 t_d^4)] = t_d^4 / t_{d1}^4 = t_d^4 / (1.2 t_d)^4 = 0.48$$

$$\text{即 } E_1 = 0.48 E$$

答：工作量可以减少到原来的 48%。

5. 已估算出 CAD 软件的代码行数为 33.5KLOC，CAD 软件为半独立型、中等规模的软件，通过查表得知 $a = 3.0$ ， $b = 1.12$ ， $c = 2.5$ ， $d = 0.35$ 。用基本 CoCoMo 模型计算开发 CAD 软件所需的工作量、开发时间以

及需要参加项目的平均人数。

$$\begin{aligned}\text{解：} \quad E &= a L^b = 3.0 \times 33.5^{1.12} = 153 \text{ PM} \\ D &= c E^d = 2.5 \times 153^{0.35} = 14.54 \text{ (月)} \\ N &= E / D = 153 / 14.54 \quad 11 \text{ 人}\end{aligned}$$

答：CAD 项目的开发工作量为 153PM，开发时间为 14.54 月，平均需要约 11 人。

6．设某软件项目由 5 个程序员组成的程序员小组来开发完成。在开发过程中每两个人之间通信的平均工作量为 0.13PM，试求开发该软件所需的通讯工作量。

解：由已知 $N=5$ ， $\mu=0.13\text{PM}$ ，开发该软件需要的通信数量为：

$$C(N) = C(5) = N(N-1)/2 = 5(5-1)/2 = 10$$

则由 5 人组成的程序员小组增加的通信工作量为：

$$E_c = \mu C(5) = 0.13\text{PM} \times 10 = 1.3\text{PM}$$

答：开发该软件所需的通讯工作量为 1.3PM。

7．要度量某软件的质量要素 F_2 （可靠性），假设 $C_{23}=0.2$ ， $C_{24}=0.3$ ， $C_{25}=0.3$ ， $C_{26}=0.2$ ，其余的 $C_{2k}=0$ ， F_2 对一致性的测量值 $M_3=0.8$ 、对容错性的测量值 $M_4=0.7$ 、对准确性的测量值 $M_5=0.6$ ，对简单性的测量值 $M_6=0.5$ ，试求可靠性的度量值 F_2 。

解：

$$\begin{aligned}F_2 &= \sum_{k=1}^{21} C_{2k} M_k \\ &= C_{23}M_3 + C_{24}M_4 + C_{25}M_5 + C_{26}M_6 \\ &= 0.2 \times 0.8 + 0.3 \times 0.7 + 0.3 \times 0.6 + 0.2 \times 0.5 \\ &= 0.65\end{aligned}$$

答：该软件可靠性 F_2 的度量值为 0.65。

8．要度量某软件的质量要素 F_2 （可靠性），假设 $C_{23}=0.1$ ， $C_{24}=0.3$ ， $C_{25}=0.4$ ， $C_{26}=0.2$ ，其余的 $C_{2k}=0$ ， F_2 对一致性的测量值 $M_3=0.7$ 、对容错性的测量值 $M_4=0.6$ 、对准确性的测量值 $M_5=0.5$ ，对简单性的测量值 $M_6=0.8$ ，试求可靠性的度量值 F_2 。

解：

$$\begin{aligned}F_2 &= \sum_{k=1}^{21} C_{2k} M_k \\ &= C_{23}M_3 + C_{24}M_4 + C_{25}M_5 + C_{26}M_6 \\ &= 0.1 \times 0.7 + 0.3 \times 0.6 + 0.4 \times 0.5 + 0.2 \times 0.8 \\ &= 0.61\end{aligned}$$

答：该软件可靠性 F_2 的度量值为 0.61。

9．要度量某软件的质量要素 F_6 （可维护性），假设 $C_{63}=0.1$ ， $C_{66}=0.2$ ， $C_{610}=0.2$ ， $C_{616}=0.1$ ， $C_{618}=0.2$ ， $C_{621}=0.2$ ，其余的 $C_{6k}=0$ ， F_6 对一致性的测量值 $M_3=0.7$ 、对简单性的测量值 $M_6=0.6$ 、对检测性的测量值 $M_{10}=0.8$ ，对简明性的测量值 $M_{16}=0.8$ ，对自文档化的测量值 $M_{18}=1$ ，对模块化的测量值 $M_{21}=0.9$ ，试求可维护性的度量值 F_6 。

解：

$$\begin{aligned}
 F_6 &= \sum_{k=1}^{21} C_{6k} M_k \\
 &= C_{63}M_3 + C_{66}M_6 + C_{610}M_{10} + C_{616}M_{16} + C_{618}M_{18} + C_{621}M_{21} \\
 &= 0.1 \times 0.7 + 0.2 \times 0.6 + 0.2 \times 0.8 + 0.1 \times 0.8 + 0.2 \times 1 + 0.2 \times 0.9 \\
 &= 0.81
 \end{aligned}$$

答：该软件可维护性 F_6 的度量值为 0.81。

10. 要度量某软件的质量要素 F_1 (正确性)，假设 $C_{11}=0.3$ ， $C_{12}=0.3$ ， $C_{13}=0.4$ ，其余的 $C_{1k}=0$ ， F_1 对可追踪性的测量值 $M_1=0.7$ 、对完全性的测量值 $M_2=0.9$ 、对一致性的测量值 $M_3=0.6$ ，试求正确性的度量值 F_1 。

解：

$$\begin{aligned}
 F_1 &= \sum_{k=1}^{21} C_{1k} M_k \\
 &= C_{11}M_1 + C_{12}M_2 + C_{13}M_3 \\
 &= 0.3 \times 0.7 + 0.3 \times 0.9 + 0.4 \times 0.6 \\
 &= 0.72
 \end{aligned}$$

答：该软件正确性 F_1 的度量值为 0.72。

11. 程序流程图如图 1 所示，试求出其巡回秩数 $V(G)$ 。

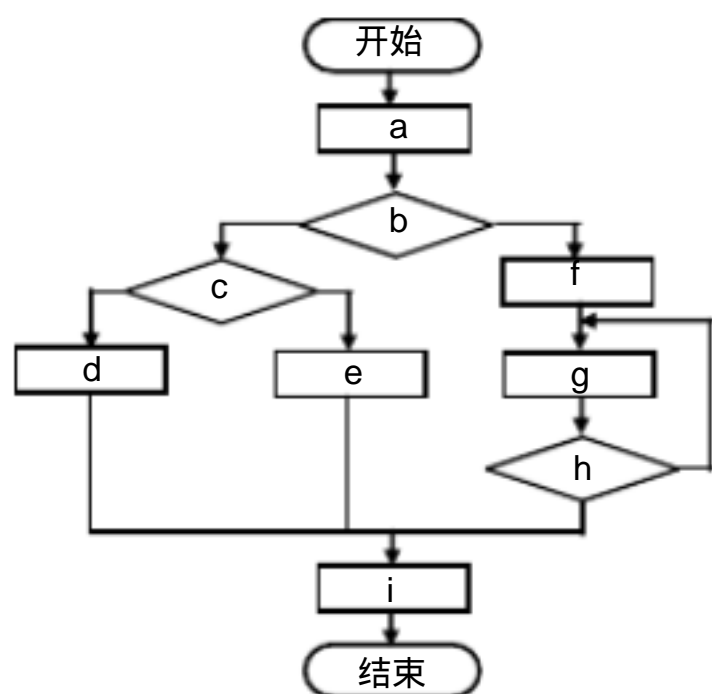


图 1 程序流程图

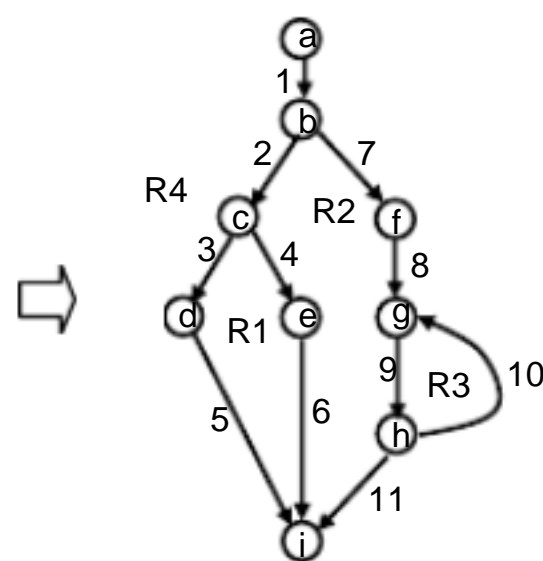


图 2 程序图

解：

(1) 画出与图 1 所示的程序流程图对应的程序图如图 2 所示。于是有：

$$V(G) = E - N + 2 = 11 - 9 + 2 = 4$$

(2) 由图 2 给出的程序图可以看出，其有界区域有 R_1 、 R_2 、 R_3 共 3 个，还有 1 个无界区域 R_4 ，共 4 个封闭区域，所以 $V(G) = 4$ 。

(3) 由图 2 给出的程序图可以看出，结点 b 、 c 、 h 是谓词结点，即谓词结点数 $P=3$ ，于是有： $V(G) = P+1 = 3+1 = 4$ 。

12. 如图 1 所示的程序流程图，试计算其巡回秩数 $V(G)$ 。

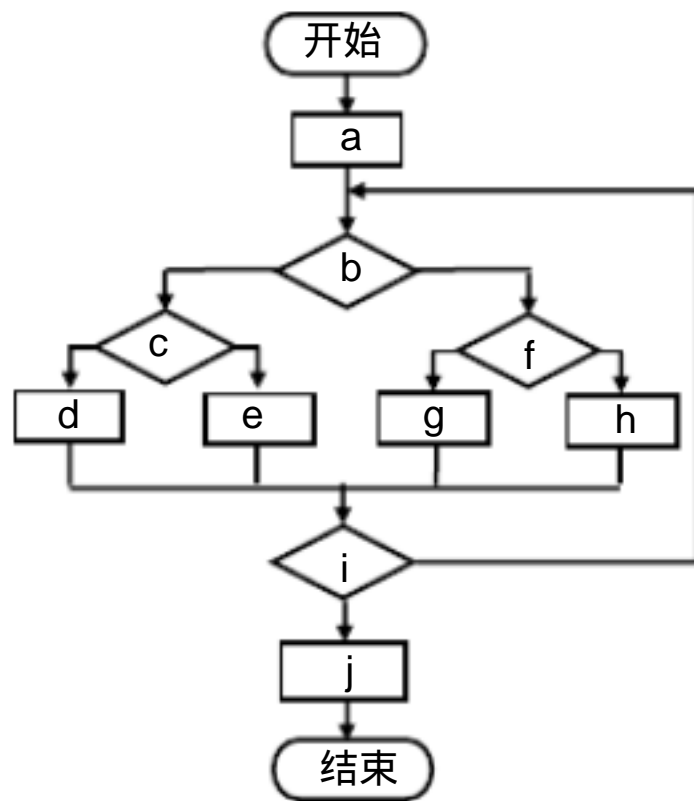


图1 程序流程图

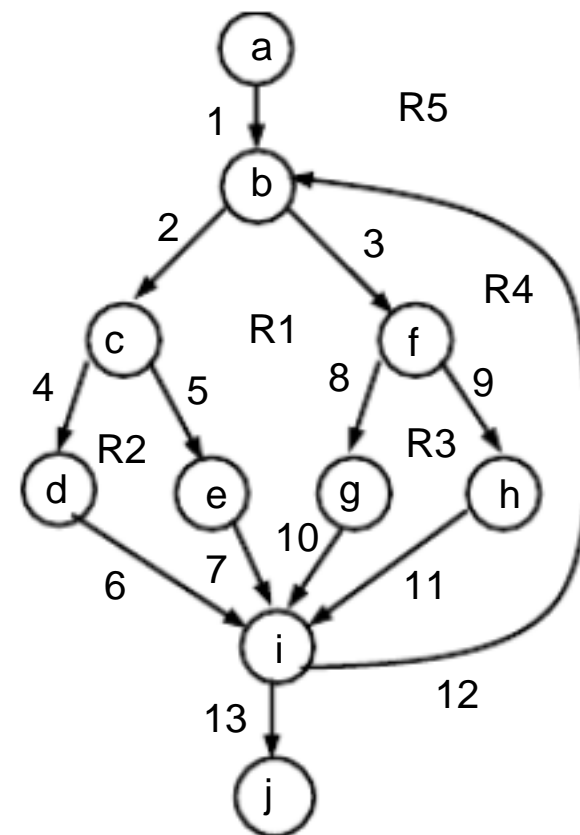


图2 程序图

解：由程序流程图画流图或称程序图如图 2 所示。于是有：

$$(1) V(G) = E - N + 2 = 13 - 10 + 2 = 5$$

$$(2) V(G) = \text{封闭区域个数} = 5$$

$$(3) V(G) = \text{谓词结点个数} + 1 = P + 1 = 4 + 1 = 5$$

13. 如图 1 所示的程序流程图，试计算其巡回秩数 $V(G)$ 。

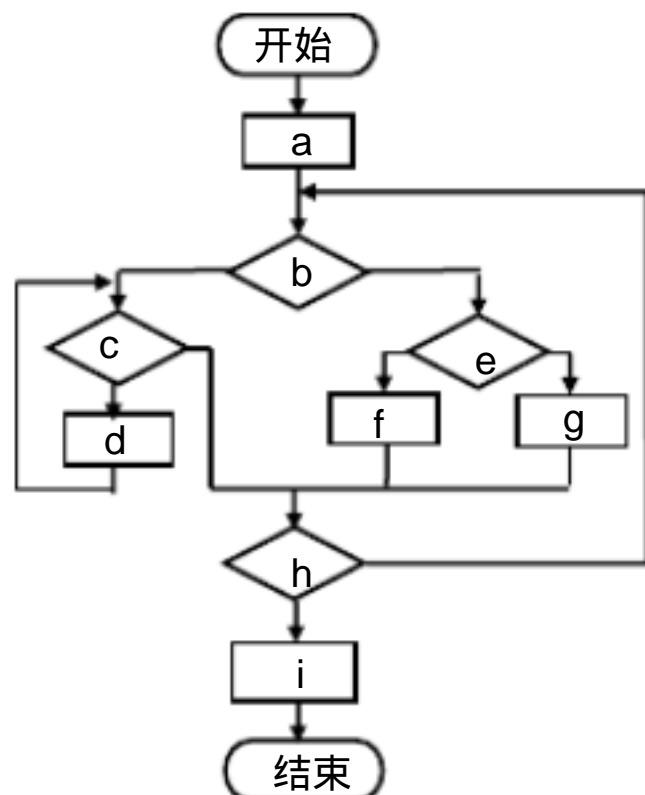


图1 程序流程图

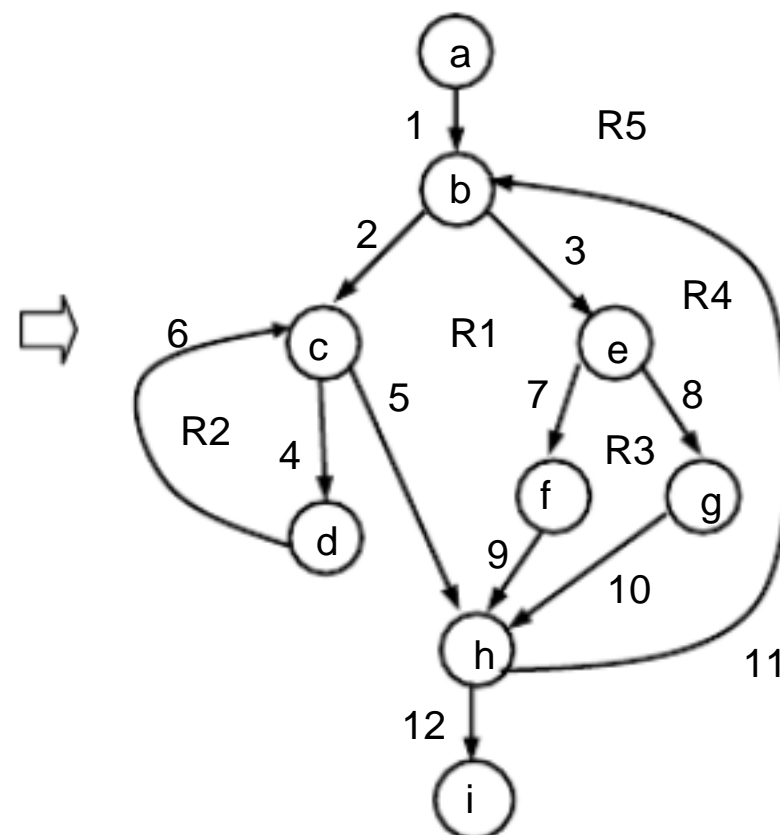


图2 程序图

解：由程序流程图画流图或称程序图如图 2 所示。于是有：

$$(1) V(G) = E - N + 2 = 12 - 9 + 2 = 5$$

$$(2) V(G) = \text{封闭区域个数} = 5$$

$$(3) V(G) = \text{谓词结点个数} + 1 = P + 1 = 4 + 1 = 5$$

14. 甲乙两名程序测试员同时对一个程序进行独立测试一个月，甲发现并改正了 24 个错误；乙发现并改正

了 20 个错误，其中有 10 个错误甲也发现了。试估算该程序经过甲乙两人这一个月测试后，还残留多少个错误？

解：由题意知， $E_1 = 24$ ， $E_2 = 20$ ， $E_0 = 10$ ，该程序所含错误总数可估算如下：

$$E_T = E_1 \times E_2 / E_0 = 24 \times 20 / 10 = 48 \text{ (个)}$$

经过一个月的测试，该程序还残留的错误数为：

$$E_r(\quad) = E_T - E_C(\quad)$$

$$E_r(1) = E_T - E_C(1)$$

$$= E_T - (E_1 + E_2 - E_0)$$

$$= 48 - (24 + 20 - 10)$$

$$= 14 \text{ (个)}$$

答：还残留 14 个错误。

15．甲乙两名程序测试员同时对一个程序进行独立测试一个月，甲发现并改正了 20 个错误；乙发现并改正了 16 个错误，其中有 8 个错误甲也发现了。试估算该程序经过甲乙两人这一个月测试后，还残留多少个错误？

解：由题意知， $E_1 = 20$ ， $E_2 = 16$ ， $E_0 = 8$ ，该程序所含错误总数可估算如下：

$$E_T = E_1 \times E_2 / E_0 = 20 \times 16 / 8 = 40 \text{ (个)}$$

经过一个月的测试，该程序还残留的错误数为：

$$E_r(\quad) = E_T - E_C(\quad)$$

$$E_r(1) = E_T - E_C(1)$$

$$= E_T - (E_1 + E_2 - E_0)$$

$$= 40 - (20 + 16 - 8)$$

$$= 12 \text{ (个)}$$

答：还残留 12 个错误。

16．甲乙两名程序测试员同时对一个程序进行独立测试一个月，甲发现并改正了 24 个错误；乙发现并改正了 23 个错误，其中有 12 个错误甲也发现了。试估算该程序经过甲乙两人这一个月测试后，还残留多少个错误？

解：由题意知， $E_1 = 24$ ， $E_2 = 23$ ， $E_0 = 12$ ，该程序所含错误总数可估算如下：

$$E_T = E_1 \times E_2 / E_0 = 24 \times 23 / 12 = 46 \text{ (个)}$$

经过一个月的测试，该程序还残留的错误数为：

$$E_r(\quad) = E_T - E_C(\quad)$$

$$E_r(1) = E_T - E_C(1)$$

$$= E_T - (E_1 + E_2 - E_0)$$

$$= 46 - (24 + 23 - 12)$$

$$= 11 \text{ (个)}$$

答：还残留 11 个错误。

17．对一个包含 10000LOC 的程序进行一个月的测试后，总共改正了 15 个错误，此时 MTBF=10h，又经过

一个月测试后，改正了 10 个错误，此时 MTBF=15h。试完成：

(1) 根据已知数据确定 MTBF 与测试时间之间的函数关系。

(2) 为做到 MTBF=100h，必须进行多长时间的测试？当测试结束时，总共改正了多少个错误？还有多少个错误潜伏在程序中？

解：

$$(1) \quad E_r(t) = k \cdot t, \text{ MTBF} = 1/k, \quad E_r(t) = I_T \cdot t \quad E_r(t) = I_T / (k \cdot \text{MTBF})$$

$$E_c(t) = E_T - I_T / (k \cdot \text{MTBF})$$

$$\text{即：} \begin{cases} 15 = E_T - 10000 / (k \cdot 10) \\ 15 + 10 = E_T - 10000 / (k \cdot 15) \end{cases}$$

解上述方程组，得： $E_T = 45, k = 100/3$

假设：单位时间内改正错误后剩余的错误数与改正前错误总数成正比，于是有：

$$E_r(1) = k_1 E_T \quad (1)$$

$$k_1 E_r(1) = k_1^2 E_T \quad (2)$$

$$\text{用数学归纳法可证明：} \quad E_r(t) = k_1^t E_T \quad (3)$$

由(1)式， $45 - 15 = k_1 \times 45$ ，得 $k_1 = 2/3$ ，

$$\text{MTBF}(t) = I_T / (k E_r(t)) = I_T / (k k_1^t E_T)$$

$$= 10000 / [(100/3) \times (2/3)^t \times 45] = 6.666667 \times (3/2)^t \quad (4)$$

(2) 将已知 MTBF(t) = 100，代入(4)式： $100 = 6.666667 \times (3/2)^t$

$$= \ln 15 / \ln 1.5 = 6.68 \text{ (月)}$$

将 $t = 6.68$ (月) 代入上面的(3)式，得：

$$E_r(6.68) = (2/3)^{6.68} \times 45 = 2.9986 \quad 3 \text{ (个)}$$

$$E_c(6.68) = E_T - E_r(6.68) = 45 - 3 = 42 \text{ (个)}$$

答：(1) MTBF 与测试时间之间的函数关系： $\text{MTBF}(t) = 6.666667 \times (3/2)^t$ 。

(2) 为做到 MTBF=100h，必须进行 6.68 (月) 的测试？当测试结束时，总共改正了 42 个错误？还有 3 个错误潜伏在程序中。

软件工程单元三（计算机系统工程）

一 单项选择

1. 下列选项中，不属于可行性研究的任务的是 (A)。

A) 概要设计 B) 经济可行性研究 C) 技术可行性研究 D) 开发方案的选择

2. 下列选项中，(C) 不属于基于计算机的系统的系统要素。

A) 人 B) 数据库 C) 车间 D) 硬件

3. 在软件工程中，不属于软件定义部分的任务是 (B)。

A) 软件验收测试计划 B) 组装测试计划 C) 需求分析 D) 软件项目计划

4. 在软件工程中，(C) 不属于软件开发部分的任务。

A) 软件总体设计 B) 单元测试计划 C) 软件经销 D) 组装测试计划

二 填空

1. 计算机系统工程包括 _____、软件工程、人机工程、数据库工程等 4 个子工程。

答案：硬件工程

2. 计算机系统工程包括硬件工程、_____、人机工程、数据库工程等 4 个子工程。

答案：软件工程

3. 计算机系统工程包括硬件工程、软件工程、_____、数据库工程等 4 个子工程。

答案：人机工程

4. 计算机系统工程包括硬件工程、软件工程、人机工程、_____等 4 个子工程。

答案：数据库工程

5. 硬件工程可以划分为_____、硬件设计、硬件制造与销售维修三个阶段。

答案：硬件定义

6. 硬件工程可以划分为硬件定义、_____、硬件制造与销售维修三个阶段。

答案：硬件设计

7. 硬件工程可以划分为硬件定义、硬件设计、_____三个阶段。

答案：硬件制造与销售维修

8. 可行性研究要在_____之后进行。

答案：初步的需求定义

9. 技术可行性研究是根据客户提出的系统功能、性能要求及实现系统的各项约束条件，从_____的角度研究实现系统的可行性。

答案：技术

10. _____、原型建造和模拟是基于计算机系统技术可行性研究的有效工具。

答案：数学建模

11. 数学建模、_____和模拟是基于计算机系统技术可行性研究的有效工具。

答案：原型建造

12. 数学建模、原型建造和_____是基于计算机系统技术可行性研究的有效工具。

答案：模拟

三 名词解释

1. 计算机系统工程

答：计算机系统工程是用工程、科学和数学的原则与方法研制基于计算机的系统的有关技术方法和过程。

四 简答题

1. 在基于计算机的系统中一般都包含哪些系统要素？

答：包含的系统要素：硬件；软件；人；文档；数据库；过程。

2. 在硬件工程中，硬件定义阶段的任务有哪些？

答：硬件定义阶段的任务是：

- 1) 制定硬件开发计划，确定项目成本和工程进度；
- 2) 进行硬件需求分析，给出硬件规格说明。
- 3) 软件产品的质量往往是靠不住的。

3. 在硬件工程中，硬件设计阶段的任务有哪些？

答：硬件设计阶段的任务是：

- 1) 设计分析，画出设计图；
- 2) 必要时建造原型（即样机）并对其进行测试；
- 3) 制造分析，画出生产图。

4. 在硬件工程中，硬件制造与销售维护阶段的任务是什么？

答：硬件制造与销售维护阶段的任务是：按照质量保证计划生产硬件产品并销售，相应的服务机构对硬件产品进行售后服务。

5. 在人机工程中，人机界面的开发过程主要包括哪几个步骤？

答：1) 活动分析。

2) 动作定义和设计。

3) 动作的实现。

4) 用户环境设计。

5) 原型设计。

6. 简述数据库工程完成的主要任务有哪些？

答：数据库工程完成的主要任务有：

1) 调查用户对信息和信息处理的需求，进行可行性论证、成本估算，制定项目计划。

2) 选择支持数据库系统的硬件和软件。

3) 数据库设计与实现。

4) 数据的收集与存储。

5) 运行、管理与维护。

7. 可行性研究的任务主要有哪几个方面？

答：可行性研究的任务主要有：

1) 经济可行性研究；

2) 技术可行性研究；

3) 运行操作可行性研究；

4) 法律可行性研究；

5) 开发方案的选择。

8. 简述可行性研究的步骤。

答：典型的可行性研究的主要步骤如下：

1) 复审初步分析结果；

2) 研究现有的系统；

3) 导出新系统的高层逻辑模型；

4) 导出新系统的高层物理解法；

5) 推荐建议的方案；

6) 评审、复审和决策。

9. 基于计算机系统的成本主要由哪 4 个部分组成？

答：基于计算机系统的成本主要由以下 4 个部分组成：

1) 购置并安装软硬件及有关设备的费用；

2) 系统开发费用；

3) 系统安装、运行和维护费用；

4) 人员培训费用。

10. 技术可行性研究主要包括哪些分析，各分析的任务如何？

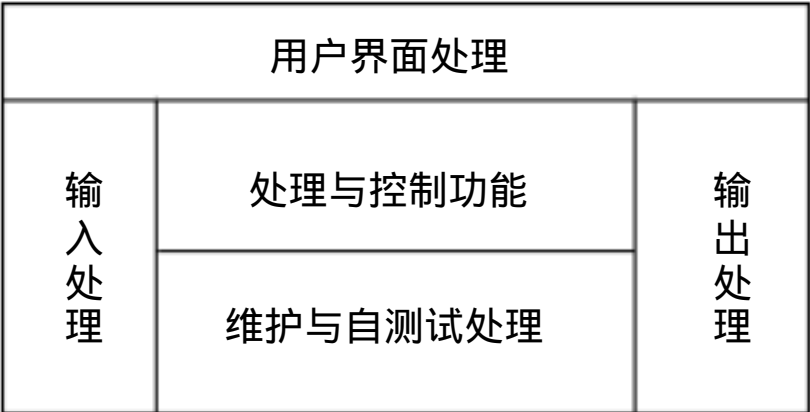
答：1) 风险分析。其任务是论证在给定的约束条件下，能否实现系统所需的功能和性能。

2) 资源分析。其任务是论证是否具备系统开发所需的各类人员、软硬件资源和工作环境等。

3) 技术分析。其任务是论证现有的技术水平和开发能力是否支持开发的全过程，并达到系统功能和性能的目标。

11. 试画出基于计算机的系统结构模板。

答：基于计算机的系统结构模板如下图所示：



12．系统建模和模拟的主要步骤有哪些？

答：系统建模和模拟的主要步骤如下：

- 1) 分析问题，确定模拟的目标。
- 2) 建立模型。
- 3) 借助模型对现实系统进行模拟，并评价模拟结果。
- 4) 如果模拟结果不正确，则修改模型，然后在 3) 和 4) 间进行迭代，直至得到满意的结果为止。
- 5) 撰写模拟文档。

13．对系统规格说明的技术评审主要解决哪些问题？

答：对系统规格说明的技术评审主要解决的问题有：

- 1) 系统规格说明中的定义是否正确，是否正确地描述了项目的范围，是否准确地定义了系统的功能、性能和界面，开发人员和用户对系统的目标是否有共同的认识等。
- 2) 系统功能的复杂性是否与开发风险、成本和进度预测保持一致。
- 3) 系统及各子系统功能定义是否足够详细。
- 4) 系统与环境及各子系统之间的接口定义是否详细、有否遗漏。
- 5) 是否指明系统性能、可靠性和可维护性等需求。
- 6) 是否为以后的开发打下坚实的基础。

14．对系统规格说明的管理复审主要解决哪些问题？

答：对系统规格说明的管理复审主要解决的问题有：

- 1) 系统是否有稳定的商业需求，是否有经济和社会效益。
- 2) 系统开发是否还有其他的选择方案。
- 3) 系统各部分开发风险如何。
- 4) 系统开发所需资源是否具备。
- 5) 成本和进度计划是否合理等。

五、综合题

1．已知一个待开发的基于计算机的系统的总成本的估算值的折现值为 10000 元，预计新系统投入运行后每年可带来 5000 元的收入，假设新系统的生存周期（不包括开发时间）为 4 年，当年的年利率为 10%，试求该系统的纯收入 T、投资回收期 D 和投资回收率 j。

N（年）	第 n 年的收入	$(1+i)^n$	折现值	累计折现值
1	5000	1.1	4545.45	4545.45
2	5000	1.21	4132.23	8677.68
3	5000	1.331	3756.57	12434.25
4	5000	1.4641	3415.07	15849.32

解：对该系统将来的收入折现，计算结果如上表所示。

- 1) 纯收入： $T = P_T - S_T = 15849.32 - 10000 = 5849.32$ （元）
- 2) 投资回收期： $D = 2 + (10000 - 8677.68) / 3756.57 = 2.352$ （年）
- 3) 投资回收率：

$$S = F_1(1+j)^{-1} + F_2(1+j)^{-2} + F_3(1+j)^{-3} + F_4(1+j)^{-4}$$

代入数据，有：

$$10000 = 5000 \times (1+j)^{-1} + 5000 \times (1+j)^{-2} + 5000 \times (1+j)^{-3} + 5000 \times (1+j)^{-4}$$

$$= 5000 \times [(1+j)^{-1} + (1+j)^{-2} + (1+j)^{-3} + (1+j)^{-4}]$$

即： $2 = (1+j)^{-1} + (1+j)^{-2} + (1+j)^{-3} + (1+j)^{-4}$

用逐次逼近法解之，得： j 34.9%

答：该系统的纯收入为 5949.32 元，投资回收期为 2.352 年，投资回收率为 34.9%。

2．已知一个待开发的基于计算机的系统的总成本的估算值的折现值为 5000 元，预计新系统投入运行后每年可带来 2500 元的收入，假设新系统的生存周期（不包括开发时间）为 5 年，当年的年利率为 12%，试求该系统的纯收入 T、投资回收期 D 和投资回收率 j。

N（年）	第 n 年的收入	$(1+i)^n$	折现值	累计折现值
1	2500	1.12	2232.14	2232.14
2	2500	1.2544	1992.98	4225.12
3	2500	1.404928	1779.45	6004.57
4	2500	1.57351936	1588.80	7593.37
5	2500	1.762341683	1418.57	9011.94

解：对该系统将来的收入折现，计算结果如上表所示。

1) 纯收入： $T = P_T - S_T = 9011.94 - 10000 = 4011.94$ （元）

2) 投资回收期： $D = 2 + (5000 - 4225.12) / 1779.45 = 2.44$ （年）

3) 投资回收率：

$$S = F_1(1+j)^{-1} + F_2(1+j)^{-2} + F_3(1+j)^{-3} + F_4(1+j)^{-4} + F_5(1+j)^{-5}$$

代入数据，有：

$$5000 = 2500 \times (1+j)^{-1} + 2500 \times (1+j)^{-2} + 2500 \times (1+j)^{-3} + 2500 \times (1+j)^{-4} + 2500 \times (1+j)^{-5}$$

$$= 2500 \times [(1+j)^{-1} + (1+j)^{-2} + (1+j)^{-3} + (1+j)^{-4} + (1+j)^{-5}]$$

即： $2 = (1+j)^{-1} + (1+j)^{-2} + (1+j)^{-3} + (1+j)^{-4} + (1+j)^{-5}$

用逐次逼近法解之，得： j 41%

答：该系统的纯收入为 4011.94 元，投资回收期为 2.44 年，投资回收率为 41%。

一、 选择题

1． 需求分析最终结果是产生（ ）

A． 项目开发计划 B． 可行性分析报告 C． 需求规格说明书 D． 设计说明书

答案： C

2． 需求分析中，开发人员要从用户那里解决的最重要的问题是（ ）

A． 让软件做什么 B． 要给软件提供哪些信息
C． 要求软件工作效率怎样 D． 让软件具有何种结构

答案： A

3． DFD 中的每个加工至少有（ ）

A． 一个输入流或一个输出流 B． 一个输入流和一个输出流
C． 一个输入流 D． 一个输出流

答案： B

4． 数据字典总，一般不出现的条目是（ ）

A 数据流 B 数据存储 C． 加工 D． 源点与终点

答案： D

5． 分层数据流图是一种比较严格又易于理解的描述方式，它的顶层图描述了系统的（ ）

A . 细节 B . 输入与输出 C . 软件的作用 D . 绘制的时间

答案： B

6 . 需求规格说明书的内容不应包括对 () 的描述。

A . 主要功能 B . 算法的详细过程 C . 用户界面和运行环境 D . 软件性能

答案： B

7 . 需求规格说明书的作用不应包括 ()

A . 软件设计的依据 B . 用户与开发人员对软件要做什么的共同理解

C . 软件验收的依据 D . 软件可行性研究的依据

答案： D

8 . SA 方法用 DFD 描述 ()

A . 系统的控制流程 B . 系统的数据结构

C . 系统的基本加工 D . 系统的功能

答案： D

9 . 一个局部数据存储当它作为 () 时，就把他画出来。

A . 某些加工的数据接口 B . 某个加工的特定输入

C . 某个加工的特定输出 D . 某些加工的数据接口或某个加工的特定输入输出

答案： D

10 . 对于分层的 DFD ，父图与子图的平衡指子图的输入、输出数据流同父图相应加工的输入、输出数据流 ()

A . 必须一致 B . 数目必须相等 C . 名字必须相同 D . 数目必须不等

答案： A

11 . 需求分析阶段不适用于描述加工逻辑的工具是 ()

A . 结构化语言 B . 判定表 C . 判定树 D . 流程图

答案： D

12 . SA 方法的分析步骤是首先调查了解当前系统的工作流程，然后 ()

A . 获得当前系统的物理模型，抽象出当前系统的逻辑模型，建立目标系统的逻辑模型

B . 获得当前系统的物理模型，抽象出当前系统的逻辑模型，建立目标系统的物理模型

C . 获得当前系统的逻辑模型，建立当前系统的物理模型，抽象出目标系统的逻辑模型

D . 获得当前系统的逻辑模型，建立当前系统的物理模型，建立目标系统的物理模型

答案： A

13 . SA 方法的基本思想是 ()

A . 自底向上逐步抽象 B . 自底向上逐步分解

C . 自顶向下逐步分解 D . 自顶向上抽象

答案： C

14 . 初步用户手册在 () 阶段编写

A . 可行性研究 B . 需求分析 C . 软件概要设计 D . 软件详细设计

答案： B

15 . 系统中模块的 ____ 不仅意味着作用于系统的小变动将导致行为上的小变化，也意味着规格说明的小变动将影响到一小部分模块。

A . 可分解性 B . 保护性 C . 可理解性 D . 连续性

答案： D

16 . 下面关于面向对象方法中消息的叙述，不正确的是 _____。

- A..键盘、鼠标、通信端口、网络等设备一有变化，就会产生消息
- B．操作系统不断向应用程序发送消息，但应用程序不能向操作系统发送消息
- C. 应用程序之间可以相互发送消息
- D．发送与接收消息的通信机制与传统的子程序调用机制不同

答案：B

17．面向对象技术中，对象是类的实例。对象有三种成份：_____、属性和方法（或操作）。

- A. 标识
- B. 规则
- C. 封装
- D. 消息

答案：A

18．可行性研究要进行一次_____需求分析。

- A．详细的
- B．全面的
- C．简化的、压缩的
- D．彻底的

答案 :C

19．系统流程图用于可行性分析中的_____的描述。

- A．当前运行系统
- B.当前逻辑模型
- C．目标系统
- D．新系统

答案 :A

20．系统流程图是描述_____的工具。

- A．逻辑系统
- B．程序系统
- C．体系结构
- D．物理系统

答案 :C

21_____工具在软件详细设计过程中不采用。

- A．判定表
- B．IPO图
- C．PDL
- D．DFD图

答案 :D

22.程序的三种基本控制结构是

- A．过程、子程序和分程序
- B．顺序、选择和重复
- C．递归、堆栈和队列
- D．调用、返回和转移

答案 :B

23．程序的三种基本控制结构的共同特点是

- A．不能嵌套使用
- B．只能用来写简单程序
- C．已经用硬件实现
- D．只有一个入口和一个出口

答案 :D

24．软件开发过程中，抽取和整理用户需求并建立问题域精确模型的过程叫

- A．生存期
- B．面向对象设计

C．面向对象程序设计

D．面向对象分析

答案 :D

25．原型化方法是用户和设计者之间执行的一种交互构成，适用于_____系统。

A．需求不确定性高的

B．需求确定的

C．管理信息

D．实时

答案 :A

26．原型化方法是一种_____型的设计过程。

A．自外向内

B．自顶向下

C．自内向外

D．自底向上

答案 :A

27.选择结构的复杂性比顺序结构的复杂性要()。

A、小 B、大 C、相等 D、无法比较

答案 :B

28.在数据流图中，(椭圆)代表()。

A、源点 B、终点 C、加工 D、模块

答案 :C

29 模块内聚度越高，说明模块内各成分彼此结合的程度越()。

A、松散 B、紧密 C、无法判断 D、相等

答案 :B

30、软件设计阶段的输出主要是()。

A、程序 B、模块 C、伪代码 D、设计规格说明书

答案 :D

31、软件需求分析是保证软件质量的重要步骤，它的实施应该是在()。

A、编码阶段 B、软件开发全过程 C、软件定义阶段 D、软件设计阶段

答案 :C

32、在七种耦合中，最低耦合是()。

A、内容耦合 B、公共耦合 C、数据耦合 D、非直接耦合

答案 :D

33、结构化分析方法 SA、结构化设计方法 SD，和 JACKSON 方法，是在软件开发过程中常用的方法，人们使用 SA 方法时可以得到()，使用 SD 方法时可以得到()。

A、程序流程图 B、具体的语言程序 C、模块结构图及模块的功能说明书

D、分层数据流图

答案 :D

34．进行需求分析可使用多种工具，但()是不适用的。

A．数据流图 B．判定表

C．PAD 图 D．数据字典

答案 :C

35．需求分析阶段研究的对象是软件项目的()。

A．用户要求 B．合理要求

C．模糊要求

答案 :C

37．数据字典的任务是对于数据流图中出现的所有被命名的数据元素，在数据字典中作为一个词条加以定义，使得每一个图形元素的名字都有一个确切的 ()。

- A．对象 B．解释
c．符号 D．描述

答案 :D

38．在数据流图中，有名字及方向的成分是 ()。

- A．控制流 B．信息流
c．数据流 D．信号流

答案 :C

39．在结构化分析方法中，用以表达系统内数据的运动情况的工具有 ()

- A．数据流图 B．数据词典
C．结构化英语 D．判定树与判定表

答案 :A

40．在结构化分析方法中，用以表达系统内数据的运动情况的工具有 ()

- A．数据流图 B．数据词典
C．结构化英语 D．判定树与判定表

答案 :B

41．结构化分析方法 (SA) 是一种面向 () 的需求分析方法。

- A 一对象 B．数据结构 C．数据流 D．控制流

答案 :C

42.描述结构化系统分析方法的工具不包括 ()

- A．数据流图 B．组织结构图
C．数据词典 D．结构化语言

答案 :B

43．决策树 ()

- A．能用来代替程序流程图
B．是程序流程图的辅助手段
C．是描述基本加工的逻辑功能的有效工具
D．A 和 B

答案 :C

44．软件需求分析阶段的工作，可以分成以下四个方面：对问题的识别、分析与综合、制定规格说明以及 ()。

- A．总结 B．实践性报告
C．需求分析评审 D．以上答案都不正确

答案 :C

45．以下说法错误的是 ()。

- A．结构化语言外层中的顺序结构是一组祈使语句、选择语句、重复语句的顺序排列
B．结构化语言外层中的选择结构使用 IF-THEN--ELSE-ENDIF 等关键字
C．结构化语言的内层可以采用祈使语句的自然语言短语
D．结构化语言外层中的重复结构使用 CASE-OF-ENDCASE 等关键字

答案 :D

46．结构化分析方法使用的描述工具 ()定义了数据流图中每一个图形元素。

- A．数据流图 B．数据字典
- c．判定表 D．判定树

答案 :B

47．下列说法正确的是 ()。

- A．对于顺序执行和循环执行的动作，用判定表和判定树
- B．对于存在多个条件复杂组合的判断问题，用结构化语言描述
- C．判定表较判定树直观易读，判定树进行逻辑验证较严格
- D．可将判定表和判定树两者结合起来，先用判定表作底稿，在此基础上产生判定树

答案 :D

48．需求规格说明书的作用不应包括 ()。

- A．软件设计的依据
- B．用户与开发人员对软件要做什么的共同理解
- c．软件验收的依据
- D．软件可行性研究的依据

答案 :D

49．软件需求规格说明书的内容不应包括对 ()的描述。

- A． 主要功能 B．算法的详细过程
- C．用户界面及运行环境 D．软件的性能

答案 :B

50．在结构化分析方法 (SA)中，与数据流图配合使用的是 ()。

- A．网络图 B．实体联系图
- C．数据字典 D．程序流程图

答案 :C

51．通过 ()可以完成数据流图的细化。

- A．结构分解 B．功能分解
- C．数据分解 D．系统分解

答案 :B

52．分层 DFD 是一种比较严格又易于理解的描述方式，它的顶层图描述了系统的 ()。

- A．细节 B．输入与输出
- c．软件的作者 D．绘制的时间

答案 :B

53．数据存储和数据流都是 ()仅仅所处的状态不同。

- A．分析结果 B．事件
- C．动作 D．数据

答案 :D

54．数据字典中，一般不包括下列选项中的 ()条目。

- A．数据流 B．数据存储
- C．加工 D．源点与终点

答案 :D

55．在软件需求分析中，开发人员要从用户那里解决的最重要的问题是 ()。

- A．要让软件做什么 B．要给该软件提供哪些信息
- c．要求软件工作效率怎样 D．要让软件具有何种结构

答案 :A

56．需求分析最终结果是产生 ()。

- A．项目开发计划
- B．可行性分析报告
- C．需求规格说明书
- D．设计说明书

答案 :C

57．SA 方法用 DFD 描述 ()。

- A．系统的控制流程
- B．系统的数据结构
- C．系统的基本加工
- D．系统的功能

答案 :D

58．SA 方法的分析步骤是首先调查了解当前系统的工作流程，然后 ()。

- A．获得当前系统的物理模型，抽象出当前系统的逻辑模型，建立目标系统的逻辑模型
- B．获得当前系统的物理模型，抽象出目标系统的逻辑模型，建立目标系统的物理模型
- C．获得当前系统的逻辑模型，建立当前系统的物理模型，抽象出目标系统的逻辑模型
- D．获得当前系统的逻辑模型，建立当前系统的物理模型，建立目标系统的物理模型

答案 :A

59．需求分析阶段不适于描述加工逻辑的工具是 ()。

- A．结构化语言
- C．判定树
- B．判定表
- D．流程图

答案 :D

60．结构化分析方法 (SA) 最为常见的图形工具是 ()。

- A．程序流程图
- B．实体联系图
- C．数据流图
- D．结构图

答案 :C

61．sA 方法的基本思想是 ()。

- A．自底向上逐步抽象
- B．自底向上逐步分解
- C．自顶向下逐步分解
- D．自顶向下逐步抽象

答案 :C

62．下面错误的说法是 ()。

- A．每个数据流必须用名词或名词短语命名
- B．每个加工必须有名字，通常是动词短语
- c．每个数据存储必须用名词或名词短语
- D．每个数据源点或终点必须有名字

答案 :C

63．下面错误的说法是 ()。

- A．判定表能够把在什么条件下系统应做什么动作准确无误地表示出来
- B．判定表能够描述循环的处理特性
- C．结构化语言同样能够描述循环的处理特性
- D．判定树是判定表的变形，一般情况下它比判定表更直观，且易于理解和使用

答案 :B

64．软件需求说明书是软件需求分析阶段的重要文件，下述 ()是其应包含的内容。

- 数据描述
- 功能描述
- 模块描述
- 性能描述

- A．
- B．

c . D .

答案 :D

65 . 软件需求分析方法中 , 结构化分析是一种常用的方法。 结构化分析产生的系统说明书是包括一套分层的()图。

- A 流程 B 数据流
C 软件结构 D 事务转换

答案 :B

66 . 软件需求分析方法中 , 结构化分析是一种常用的方法。结构化分析产生的系统说明书是包括一本 (B)。

- A 结构说明 B 用户需求
C 数据字典 D 数据流程

答案 :B

67 . 常用的需求分析方法有面向数据流的结构化分析方法 , () , 面向对象的分析方法等。

- A . 面向数据结构的分析方法
B . 面向时序问题的分析方法
C . 面向事务的分析方法

答案 :A

68 . 结构化分析方法使用的描述工具 ()描述系统由哪几部分组成 , 各部分之间有什么联系等等。

- A . 数据流图 B . 数据字典
C . 判定表 D . 判定树

答案 :A

69 . 结构化分析方法 (sA) 是一种面向 ()需求分析方法。

- A . 对象 B . 数据结构
C . 数据流 D . 结构图

答案 :C

70 . 需求分析中 , 对算法的简单描述记录在 ()中。

- A . 层次图 B . 数据字典
C . 数据流图 D . IPO 图

答案 :B

71 . 数据流图和 ()共同构成系统的逻辑模型。

- A . IPO 图 B . PDL 图
C . 数据字典 D . 层次图

答案 :B

72 . 在下面的叙述中 , ()不是软件需求分析的任务。

- A . 问题分解 B . 可靠性与安全性要求
c . 结构化程序设计 D . 确定逻辑模型

答案 :C

73 . 原型化方法是一类动态定义需求的方法 , 下列叙述中 , ()不具有原型化方法的特征。

- A . 提供严格定义的文档 B . 加强用户参与和决策
C . 简化项目管理 D . 加快需求的确定

74 . 需求分析是 ()。

- A . 软件开发工作的基础 B . 软件生存周期的开始
C . 由系统分析员单独完成的 D . 由用户自己单独完成的

答案 :A

75．软件原型化开发方法有其基本步骤，下述 ()是实施原型化的最基本的步骤。

获得基本需求

开发工作模型

严格细部说明

模型验证

A．全部都是 B． 和

c． D． 和

答案 :B

76．原型化方法是用户和软件开发人员之间进行的一种交互过程，适用于 (A)系统。

A 需求不确定性高的 B 需求确定的

C 管理信息 D 动态改变

答案 :A

77. 原型化方法从用户界面的开发人手，首先形成 (B)。

A 用户界面使用手册 B 用户界面需求分析说明书

C 系统界面原型 D 完善的用户界面

答案 :C

78 原型化方法是一种 ()型的设计过程。

A 自外向内 B 自顶向下

C 自内向外 D 自底向上

答案 :A

79．数据流图是用于软件需求分析的工具，下列元素 ()是其基本元素。

数据流 加工 数据存储 外部实体

A． 和 B． 和

c．全部 D． 和

答案 :C

80．结构化分析方法是一种预先严格定义需求的方法，它在实施时强调的是分析对象的

()。

A．控制流 B．数据流

C．程序流 D．指令流

答案 :A

81． 软件开发常使用的两种基本方法是结构化和原型化方法，在实际应用中，它们之间的关系常表现为

()。

A．互相排斥 B．互相补充

C．独立使用 D．交替使用

答案 :B

82． 软件工程的结构化生命周期方法 (sA)是将软件开发的全过程划分为互相独立而又互相依存的阶段，软件的逻辑模型是形成于 ()。

A 开发阶段 B 计划阶段

C 分析阶段 D 设计阶段

答案 :C

83． 作为软件开发结构化方法的工具， 数据流图描述数据在软件流动和被处理变换的过程， 它是以图示的方法来表示 ()。

A 软件模型 B 软件功能

C 软件结构 D 软件加工

答案 :C

84 . 数据流图的正确性是保证软件逻辑模型正确性的基础，以下 ()项内容与它的正确性、相关性最弱。

A 数据守恒 B 均匀分解
C 文件操作 D 图形层次

答案 :C

85 . 数据字典最基本的功能是 ()。

A 数据库设计 B 数据通讯 C 数据定义 D 数据维护

答案 :C

86 . 软件的结构化设计 (sD)方法中，一般分为总体设计和详细设计两阶段，其中总体设计主要是建立 ()。

A 软件结构 B 软件流程
C 软件模型 D 软件模块

答案 :A

87 . 结构化生命周期方法具有各种特征，下列各种叙述中， ()不是结构化方法的特征。

A . 严格定义需求 B . 划分开发阶段 C . 提供运行模型 D . 制定规范文档

答案 :C

88 . SA 方法是一种 ()。

A . 自顶向下逐层分解的分析方法
B . 自底向上逐层分解的分析方法
C . 面向对象的分析方法
D . 以上都不是

答案 :A

89 . 结构化程序流程图中一般包括三种基本结构，下述结构中 ()不属于其基本结构。

A . 顺序结构 B . 条件结构
C . 选择结构 D . 嵌套结构

答案 :D

90 . 结构化分析方法以数据流图、 ()和加工说明等描述工具，即用直观的图和简洁语言来描述软件系统模型。

A . DFD 图 B . PAD 图
C . IPO 图 D . 数据字典

答案 :D

91 . 结构化分析 (SA)是软件开发需求分析阶段所使用的方法， ()不是 SA 所使用的工具。

A . DFD 图 B . PAD 图 c . 结构化英语 D . 判定表

答案 :B

92 . Jackson 设计方法是由英国的 M . Jackson所提出的。它是一种面向 ()的软件设计方法。

A . 对象 B . 数据流
C . 数据结构 D . 控制结构

答案 :C

93 . 软件设计中， Jackson 方法是一种面向数据结构的设计方法，它将数据结构表示为

三种基本结构，分别为（ ）。

- A．分支结构、选择结构和控制结构
- B．顺序结构、选择结构和循环结构
- c．顺序结构、分支结构和嵌套结构
- D．顺序结构、选择结构和重复结构

答案 :B

94．软件工程方法是在实践中不断发展的方法，而早期的软件工程方法是指（ ）。

- A．原型化方法
- B．结构化方法
- C．面向对象方法
- D．功能分解方法

答案 :B

95．在结构化方法中，软件功能分解应属于软件开发中的（ ）阶段。

- A 详细设计
- B 需求分析
- C 总体设计
- D 编程调试

答案 :B

96．数据流程图 (DFD) 是用于描述结构化方法中（ ）阶段的工具。

- A 可行性分析
- B 需求分析
- C 详细设计
- D 程序编码

答案 :B

97．数据字典 (DD) 是定义（ ）系统描述工具中的数据的工具。

- A 数据流程图
- B 系统流程图
- C 程序流程图
- D 软件结构图

答案 :A

98．判定树和判定表是用于描述结构化分析方法中（ ）环节的工具。

- A 功能说明
- B 数据加工
- C 流程描述
- D 结构说明

答案 :B

99．软件需求分析一般应确定的是用户对软件的（ ）。

- A．功能需求
- B．非功能需求
- c．性能需求
- D．功能需求和非功能需求

答案 :D

100．数据字典是对数据定义信息的集合，它所定义的对象都包含于（ ）。

- A．数据流图
- B．程序框图
- c．软件结构
- D．方框图

答案 :A

二、 填空

1．需求分析阶段产生的最重要的文档是（ ）。

解：需求分析说明书

2．需求分析的主要任务是（ ）。

解：要回答“软件必须做什么？”

3．为解决一个复杂的问题，往往采取的策略是（ ）。

解：分解

4．SA 方法中使用半形式化的描述方式表达需求，采用的主要描述工具是（ ）。

解：数据流图和数据字典

5．数据流图中有四种符号元素，他们是（ ）。

解：数据流，加工，数据存储，实体

6．数据字典中有四类条目，分别是（ ）。

解：加工说明，数据流条目，数据存储条目，数据项条目

7．在画分层数据流图时，父图与子图的输入输出要（ ）。

解：守恒

8．需求分析阶段，分析人员要确定对问题的综合需求，其中最主要的是（ ）需求。

解：功能需求

9．结构化分析的基本思想是采用（ ）的方法，能有效的控制系统开发的复杂性。

解：自顶向下，逐步求精

10．当数据流图中的某个加工的一组动作存在着多个条件复杂组合的判断时，其加工逻辑使用（ ）。

解：判定表或判定树

11．科学工程计算需要大量的标准库函数，以便处理复杂的数值计算，可供选择的语言有：（ ）、PASCAL 语言、C 语言和 PL/1 语言。

解：FORTRAN 语言

12．详细设计的任务是确定每个模块的内部特性，即模块的算法、（ ）。

解：使用的数据

13．在 JSP方法中解决结构冲突的具体办法是（ ）。

解：中间数据结构或中间文件

14．汇编语言是面向（ ）的，可以完成高级语言无法完成的特殊功能，如与外部设备之间的一些接口工作。

解：机器

15．系统流程图是描述物理模型的传统工具，用图形符号表示系统中各个元素表达了系统中各种元素之间的（ ）情况。

解：信息流动

16．数据流图中的箭头表示（ ）。

解：数据流

17．系统流程图是描述物理模型的传统工具，用图形符号表示系统中各个元素表达了系统中各种元素之间的（ ）情况。

解：信息流动

18．数据项是指（ ）数据单元。

解：不可再分解的

19．结构化分析方法的分析策略是（ ）。

解：自顶向下逐步求精

20．Jackson 方法是一种面向（ ）的设计方法。

解：数据结构

21.软件质量保证应在（ ）阶段开始。

解：需求分析

22．在结构化分析中，用于描述加工逻辑的主要工具有三种，即：结构化语言、判定表、（ ）

解：判定树

3 0．结构化语言是介于自然语言和（ ）之间的一种半形式语言。

解：形式语言

3 1 .需求分析中，开发人员要从用户那里解决的最重要的问题是（ ）。

解：要让软件做什么

3 2．JSP 方法主要用于规模（ ）的数据处理系统。

解：不大

3 3 .JSP方法不明确的划分（ ）两个阶段。

解：软件概要设计和详细设计

3 4 .JSP方法适用于输入数据和输出数据之间有（ ）的问题求解。

解：对应关系

3 5 .快速原型模型的主要特点之一是（ ）

解：及早提供工作软件

36.需求分析的基本任务是准确的回答

解系统必须做什么）。

37.需求分析阶段研究的对象是软件项目的

解（用户要求）。

3.8 .结构化分析方法就是（ ）自顶向下逐步求精进行需求分析的方法。需求分析的目的之一就是和数据流和数据存储定义到（ ）。

解：面向数据流、元素级

39 .数据流图的基本符号包括（ ）（ ）（ ）（ ）。

解：数据输入的源点和数据输出的汇点、加工、数据流、数据存储文件

40.数据流图和（ ）共同构成系统的逻辑模型。

解：数据字典

41.数据字典的内容包括六项：（ ）（ ）（ ）（ ）（ ）（ ）

解：（数据流）、（数据项）、（数据结构）、数据存储、处理逻辑、外部实体。

42.在需求分析阶段常用的图形工具有（ ）（ ）（ ）。

解：层次方框图、Warnier图、IPO图

42.需求分析应交付的主要文档是（ ）。

解：需求规格说明书

43.层次结构的上一层是下一层的（ ），下一层是上一层的（ ）。

解：抽象、求精

44.JSD是一种典型的（ ）的分析设计方法。

解：面向数据结构

45.Jackson图除了可以表达（ ）外，还可以表达数据结构。

解：程序结构

46 .原型化方法是用户和设计者之间执行的一种交互构成，适用于（ ）系统。解：需求不确定性高的

47 .原型化方法是一种（ ）型的设计过程。

解：自外向内

48 .在JSP方法中解决结构冲突的具体办法是（ ）。

解：中间数据结构或中间文件

三、名词解释

1 .需求分析

解开发人员要准确理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的形式主义功能规约（需求规格说明）的过程。

2 .判定表

解：又称判断表，是一种图形工具，适合于描述加工判断的条件较多，各条件又相互组合的逻辑功能，它

共分四大部分：条件、状态、决策方案和决策规则。

3. 结构化分析方法

解：结构化的含义是用一组规范的步骤、准则和工具来进行某项工作。

结构化方法是把整个系统开发过程分为若干阶段，每个阶段进行若干活动，每项活动应用一系列标准、规范、方法和技术，完成一个或多个任务，形成符合规范的产品。

4. 数据流图

解：简称 DFD, 是 SA(结构化分析) 方法中用于表示系统逻辑模型的一种工具。是一种功能模型。

5. 数据字典

解：数据字典：简称 D D，就是用来定义数据流图中的各个成分具体含义的，它以一种准确的、无二义性的说明方式为系统的分析、设计及维护提供了有关元素的一致的定义和详细的描述。

6. JSP 方法

解：是面向数据结构的设计方法，其定义了一组以数据结构为指导的映射过程，它根据输入，输出的数据结构，按一定的规则映射成软件的过程描述，即程序结构。

四、简答题

1. 什么是需求分析？

答：开发人员要准确理解用户的要求，进行细致的调查分析，将用户非形式的需求陈述转化为完整的需求定义，再由需求定义转换到相应的形式化功能规约（需求规格说明）的过程。

2. 需求分析阶段的基本任务是什么？

答：需求分析阶段的基本任务是：

(1. 问题识别：

双方对问题的综合需求： a. 功能需求 b. 性能需求 c. 环境需求 d. 用户界面需求。

(2. 分析与综合，导出软件的逻辑模型。

(3. 编写文档

3. 什么是结构化方法？

解：结构化的含义是用一组规范的步骤、准则和工具来进行某项工作。

结构化方法是把整个系统开发过程分为若干阶段，每个阶段进行若干活动，每项活动应用一系列标准、规范、方法和技术，完成一个或多个任务，形成符合规范的产品。

4. 结构化方法的原则是什么？

答：其主要原则为：

(1) 用户参与的原则

(2) “先逻辑，后物理”的原则

(3) “自顶向下”的原则

(4) 工作成果描述标准化原则

5. 结构化方法使用什么描述工具？

答：a. 数据流图 b. 数据字典 c. 描述加工逻辑的结构化语言、判定表、判定树。

6. 简述结构化分析方法的步骤。

答：结构化分析方法步骤：

a. 了解当前系统的工作流程，获得当前系统的物理模型

b. 抽象出当前系统的逻辑模型。

c. 建立上标系统的逻辑模型。

d. 作进一步补充和优化。

7. 什么是数据流图？

答：数据流图：简称 DFD, 是 SA(结构化分析) 方法中用于表示系统逻辑模型的一种工具。是一种功能模型。

8. 数据流图的作用是什么？

答：它以图形的方式描绘数据在系统中流动和处理的过程，反映系统必须完成的逻辑功能。

9. 数据流图由哪些基本符号组成？

答：基本符号有四种：

->, 箭头, 表示数据流；(), 圆或椭圆, 表示加工；=, 双杠, 表示数据存储；||, 方框, 表示数据的源点或终点。

10. 什么是数据字典？

答：数据字典：简称 D D，就是用来定义数据流图中的各个成分具体含义的，它以一种准确的、无二义性的说明方式为系统的分析、设计及维护提供了有关元素的一致的定义和详细的描述。

11. 数据字典的作用是什么？

答：1) 为系统的分析、设计及维护提供了有关元素的一致的定义和详细的描述。

2) 为分析人员查找数据流图中有关名字的详细定义而服务的。

3) 它和数据流图共同构成了系统的逻辑模型，是需求规格说明书的主要组成部分。

12. 画数据流图应该注意什么事项？

答：画数据流图注意事项：

a. 命名. 不论是数据流、数据存储还是加工，合适的命名使人们易于理解其含义。

b. 画数据流而不是控制流。

c. 一般不画物质流。

d. 每个加工至少有一个输入数据流和一个输出数据流，反映出此加工数据的来源与加工的结果。

e. 编号。

f. 父图与子图的平衡。

g. 局部数据存储。

h. 提高数据流图的易理解性。

13. 数据字典有哪些条目？

答：数据流、数据项、数据存储、基本加工。

14. 描述加工逻辑有哪些方法？

答：a. 结构化语言 b. 判定表 c. 判定树

15. 简述 SA 方法的优点。

答：1) 公认的、有成效的、技术成熟、使用广泛的一种方法，比较适合于开发数据处理类型软件的需求分析。

2) 该方法利用图形等半形式化工具表达需求，简明、易读，也易于使用，为后一阶段的设计、测试、评价提供了有利条件。

16. 简述 SA 方法的缺点。

答：1) 传统的 SA 方法主要用于数据处理方面的问题，主要工具 DFD 体现了系统“做什么”的功能，但它仅是一个静态模型，没有反映处理的顺序，即控制流程。因此，不适合描述实时控制系统。

2) SA 方法在使用 DFD 在分析与描述“数据要求”方面是有局限的。

3) DFD 不适合描述人机界面系统的要求。

4) 为了更精确地描述软件包需求，提高软件系统的可靠性、安全性，也便于实现自动化，SA 方法可与形式方法结合起来。

17. 什么是基本加工？

解：数据流图中所有不进一步分解的加工，称为基本加工。描述表达基本加工逻辑功能的结构化工具主要有：

18. 描述表达基本加工逻辑功能的结构化工具有哪些？

解：(1) 自然语言的文字叙述 (2) 结构化语言 (3) 决策树 (4) 决策表 (5) 数学公式 (6) 上述各工具的联合使用

19. 描述表达基本加工逻辑功能的结构化工具有哪些特点？

解：其特点如下所述：

(1)自然语言语义丰富，语法灵活，可描述十分广泛而复杂的问题，表达人们丰富的感情和智慧。

(2)结构化语言没有严格的语法规则，使用的词汇比形式化的计算机语言广泛，但使用的语句类型很少，结构规范，表达的内容清晰、准确、易理解，不易产生歧义。

(3)决策树清晰地表达了在什么情况下应采取什么策略，不易产生逻辑上的混乱。是描述基本加工的逻辑功能的有效工具。

(4)决策表将比较复杂的决策问题简洁、明确、一目了然地描述出来。它是描述条件比较多的决策问题的有效工具。

20. 需求分析与软件设计二个阶段任务的主要区别是什么？

答：需求分析定义软件的用户需求，即定义待开发软件能做什么

软件设计定义软件的实现细节以满足用户需求，即研究如何实现软件。

21. 为什么数据流程图要分层？

答：为了表达数据处理过程的数据加工情况，用一个数据流程图是不够的。为表达稍为复杂的实际问题，需要按照问题的层次结构进行逐步分解，并以分层的数据流程图反映这种结构关系。

22. 什么是加工逻辑？

答：加工逻辑就是处理逻辑，加工逻辑是对加工过程的抽象描述。加工逻辑的具体实现就是加工过程。加工逻辑是指用户对这个加工的逻辑要求，即加工的输入数据流与输出数据流之间的逻辑关系。加工逻辑主要是说明加工做什么，而不是描述具体的加工过程，如采用哪些加工单元、如何控制执行等实现细节。

23. 怎样建立目标系统的逻辑模型？

答：在理解当前系统“怎样做”的基础上，抽取其“做什么”的本质。在物理模型中有许多物理因素，但随着分析工作的深入，有些非本质因素就显得不必要了，因而需要对物理模型进行分析，区分本质和非本质因素，去掉那些非本质因素就可获得反应系统本质的逻辑模型。做法是：

?确定目标系统与当前系统的逻辑差别；

?将变化部分看作是新的处理步骤，对功能图及对象图进行调整；

?由外及里对变化的部分进行分析，推断其结构，获得目标系统的逻辑模型。

24 什么是 . 结构化分析？

答：结构化分析方法是面向数据流进行需求分析的方法。结构化分析方法使用数据流程图 DFD 与数据字典 DD 来描述，面向数据流问题的需求分析适合于数据处理类型软件的需求描述。其核心思想是分解化简问题，将物理与逻辑表示分开，对系统进行数据与逻辑的抽象。

25. 结构化分析的“结构化”体现在哪里？

结构化分析方法体现在用抽象模型的概念，按照软件内部数据传递、变换的关系，自顶向下逐层分解，直到找到满足功能要求的所有可实现的软件为止。

26. 需求说明书由哪些部分组成？各部分之间的关系是什么？

答：软件需求说明书一般包括如下内容：

1) 引言部分 编写目的；项目背景 (应包括： a.项目的委托单位、开发单位和主管部门； b. 该软件系统与其他系统的关系。)；定义；(列出文档中所用到的专门术语的定义和缩写词的原文。)参考资料。

2) 任务概述 目标；运行环境；条件与限制。

3) 数据描述 静态数据； 动态数据 (包括输入数据和输出数据)；数据库描述 (给出使用数据库的名称和类型)；数据词典；数据采集。

4) 功能要求 功能划分；功能描述。

5) 性能需求 数据精确度；时间特性 (如响应时间、更新处理时间、数据转换与传输时间、运行时间等)；适应性 (在操作方式、运行环境、与其他软件的接口以及开发计划等发生变化时，应具有适应能力。)

6) 运行需求 用户界面 (如屏幕格式、报表格式、菜单格式、输入输出时间等)；硬件接口；软件接口；

故障处理。

7) 其他要求 如可使用性、安全保密、可维护性、可移植性等。

8) 附录

现细节。

27. 结构化设计方法的基本思想是什么？

答：结构化设计方法的基本思想是将系统划分成一些独立的功能模块，这些模块按照一定的组织层次构造起来形成软件结构，通过自顶向下逐步细化的方法将用数据流图表示的信息转换成程序结构的设计描述。

28. 结构化设计方法如何与 SA 方法相衔接？

SA 是结构化分析方法的简称。 结构化分析方法是面向数据流进行需求分析的方法。 其核心思想是分解化简问题，将物理与逻辑表示分开， 对系统进行数据与逻辑的抽象。 结构化分析方法体现在用抽象模型的概念，按照软件内部数据传递、变换的关系，自顶向下逐层分解，直到找到满足功能要求的所有可实现的软件为止。

结构化设计是在结构化分析的基础上完成的。

结构化设计属于面向数据流的设计方法。在需求分析阶段，通过 SA 方法，解决了一个关键问题：信息流。数据流是软件开发人员考虑问题的出发点和基础。数据流从系统的输入端向输出端流动，要经历一系列的变换或处理。用来表现这个过程的数据流图 (DFD)：实际上就是软件系统的逻辑模型。结构化设计要解决的任务，就是在上述需求分析的基础上，将 DFD 图映射为软件系统的结构。换句话说，这类设计方法允许把用 DFD 图表示的系统逻辑模型方便地转换成对于软件结构的初始设计描述。

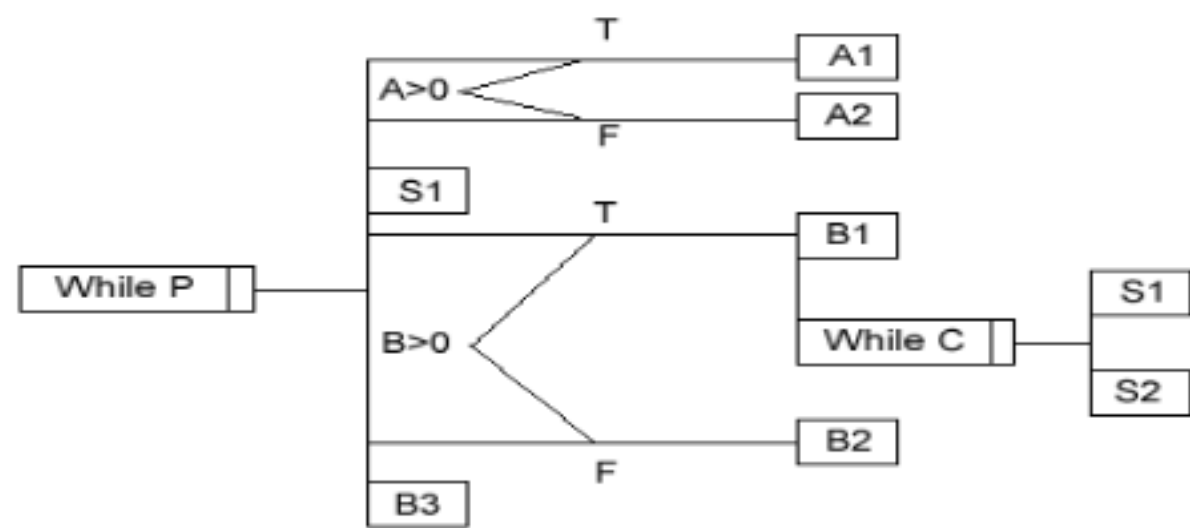
从结构化分析到结构化设计工具的转变：

结构化分析结果	结构化设计结果
数据流图	初始结构

五．综合题

画出下面用 PDL 写出的程序的 PAD 图。

```
WHILE P DO
    IF A > 0 THEN A1 ELSE A2 ENDIF;
    S1;
    IF B > 0 THEN B1;
        WHILE C DO S2;S3 ENDWHILE;
    ELSE B2
    ENDIF;
    B3
ENDWHILE;
```



1. 请用判定表画出以下问题的行为逻辑。 人们往往根据天气情况决定出门时的行装；天气可能下雨，也可能不下雨；天气可能变冷，也可能不变冷。如果天气要下雨，出门时带上雨伞；如果天气变冷，出门时要穿上大衣。

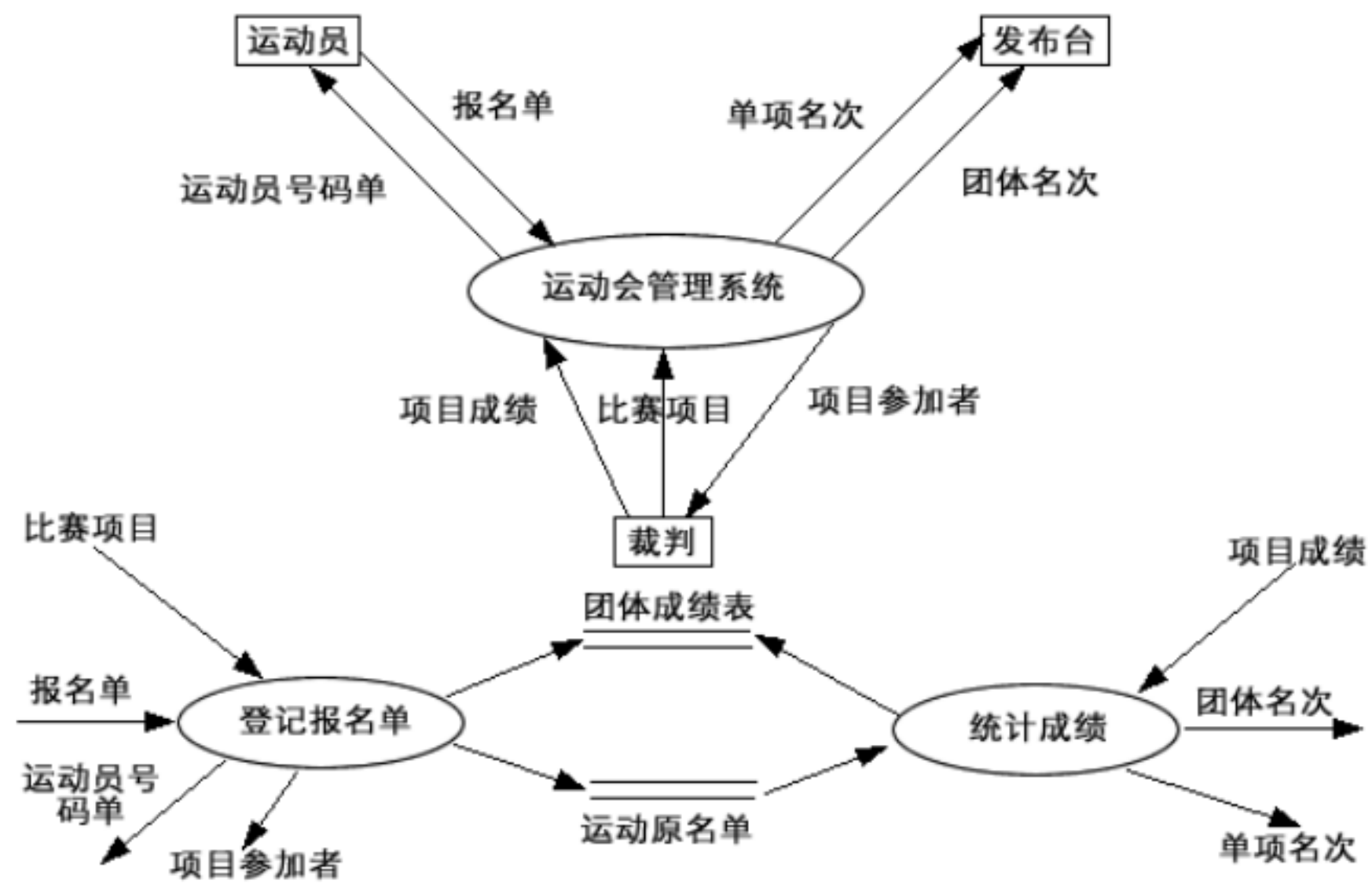
天气情况	下雨		不下雨	
	变冷	不变冷	变冷	不变冷
带雨伞	Yes	Yes	No	No
穿大衣	Yes	No	Yes	No

3. 用 SA 方法画出下列问题的顶层和 0 层数据流图。

某运动会管理系统接受来自运动员的报名单、裁判的比赛项目及项目成绩，产生运动员号码单发送给运动员，项目参加者发送给裁判，单项名次、团体名次发送给发布台。该系统有两部分功能：

（1）登记报名单：接受报名单、比赛项目，产生运动员号码单、项目参加者，形成运动员名单及团体成绩表两种数据存储。

（2）统计成绩：接受项目成绩，查询运动员名单，产生单项名次，填写团体成绩，最后产生团体名次。

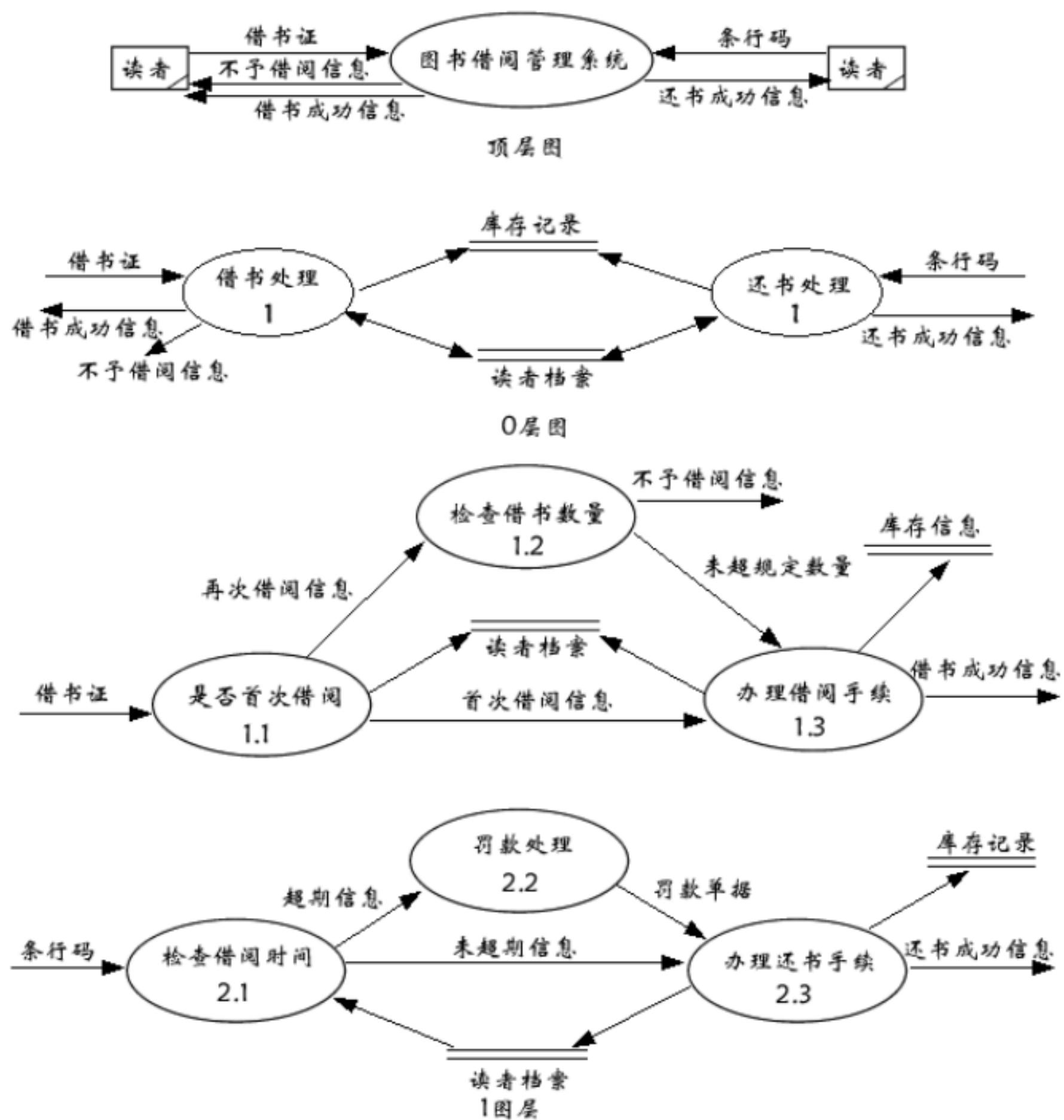


4.某图书馆借阅系统有以下功能：

（1）借书：根据读者的借书证查询读者档案，若借书数目未超过规定数量，则办理借阅手续（修改库存记录及读者档案），超过规定数量者不予借阅。对于第一次借阅者则直接办理借阅手续。

（2）还书：根据读者书中的条形码，修改库存记录及读者档案，若借阅时间超过规定期限则罚款。

请对以上问题，画出分层数据流图。

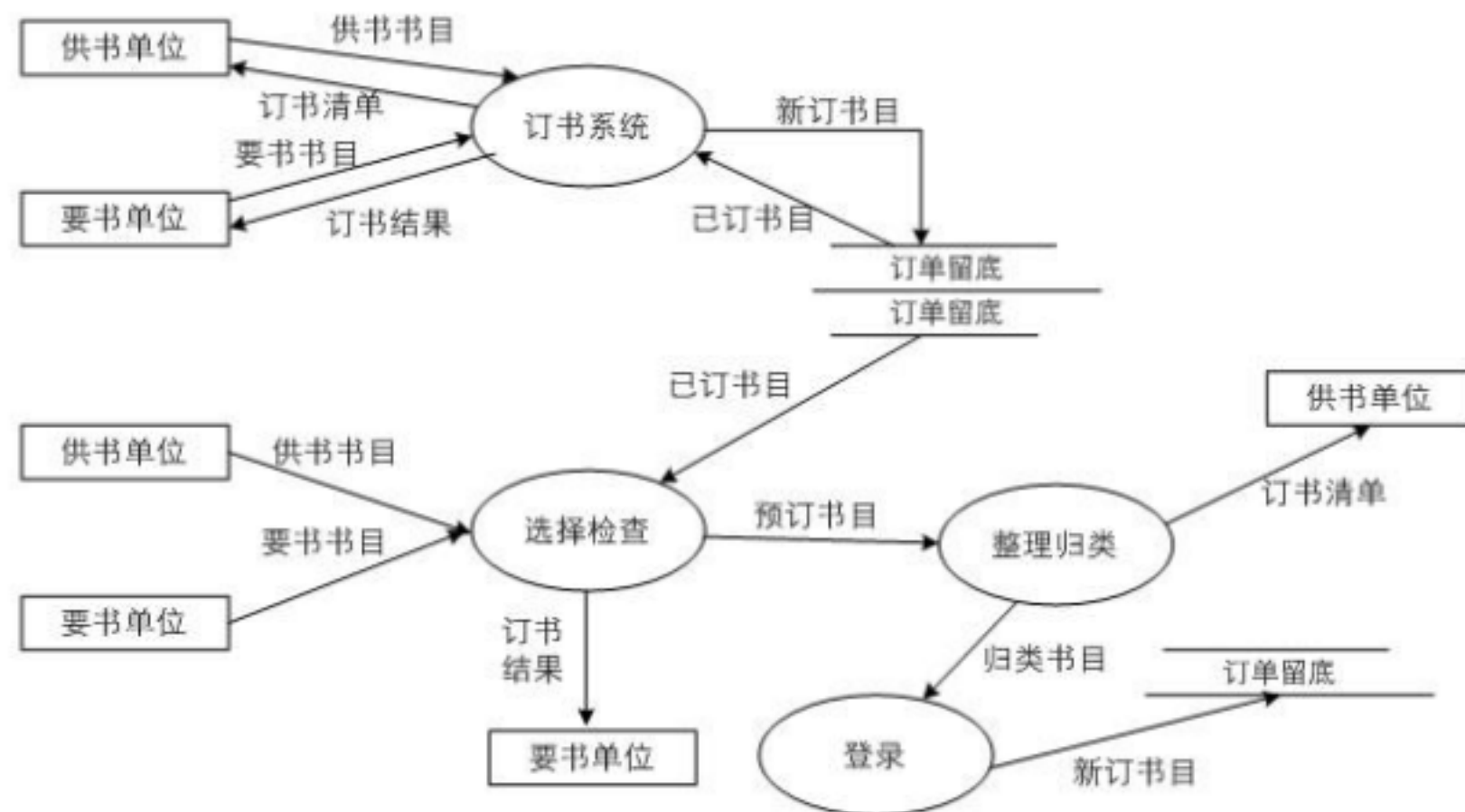


5. 图书馆的预定图书子系统有如下功能：

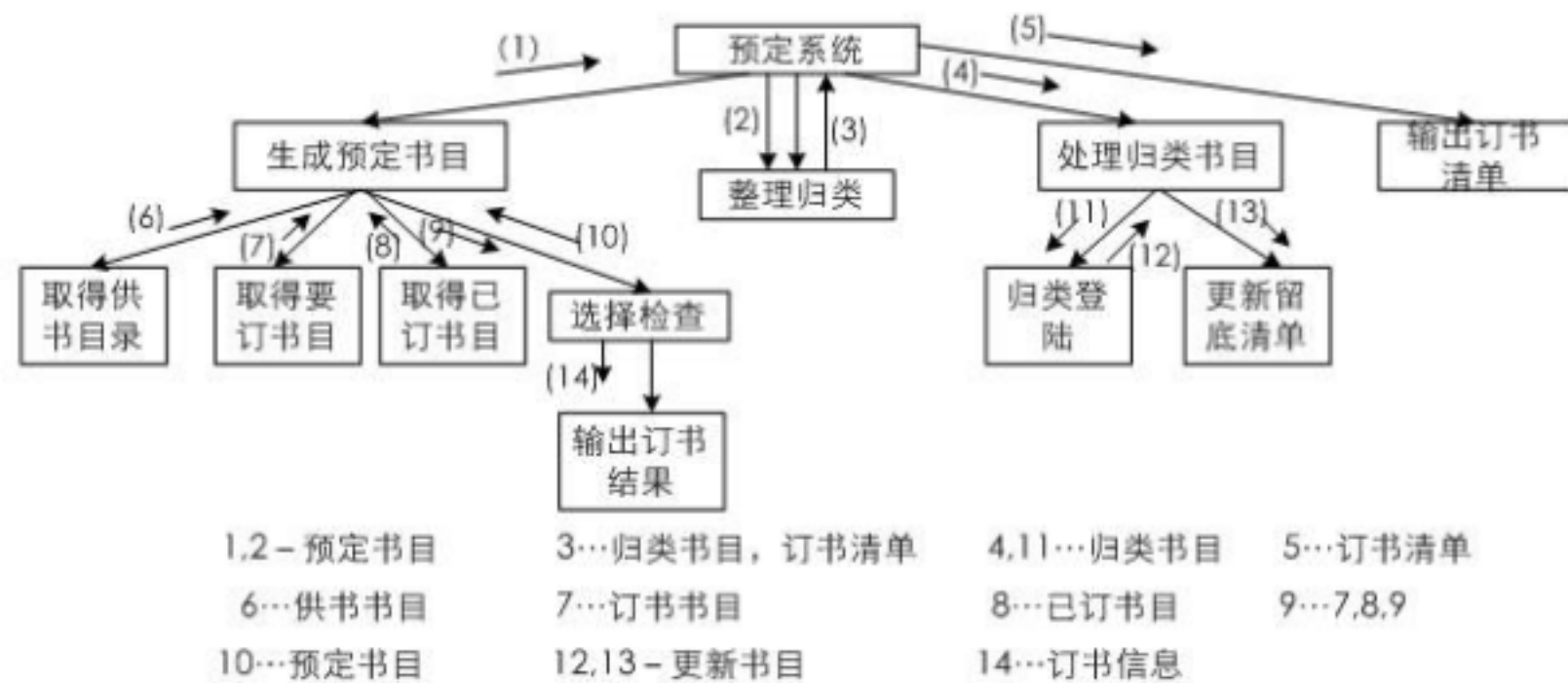
- (1) 由供书部门提供书目给订购组；
- (2) 订书组从各单位取得要订的书目；
- (3) 根据供书目录和订书书目产生订书文档留底；
- (4) 将订书信息（包括数目，数量等）反馈给供书单位；
- (5) 将未订书目通知订书者；
- (6) 对于重复订购的书目由系统自动检查，并把结果反馈给订书者。

试根据要求画出该问题的数据流程图，并把其转换为软件结构图。

解：(1) 数据流图



(2) 软件结构图



6. 某旅馆的电话服务如下：

可以拨分机号和外线号码。分机号是从 7201 至 7299。外线号码先拨 9，然后是市话号码或长话号码。长话号码是以区号和市话号码组成。区号是从 100 到 300 中任意的数字串。市话号码是以局号和分局号组成。局号可以是 455,466,888，552 中任意一个号码。分局号是任意长度为 4 的数字串。

要求：写出在数据字典中，电话号码的数据条目的定义即组成。

解：电话号码 = 分机号 | 外线号码

分机号 = 7201...7299

外线号码 = 9 + [市话号码 | 长话号码]

长话号码 = 区号 + 市话号码

区号 = 100...300

市话号码 = 局号 + 分局号

局号 = [455 | 466 | 888 | 552]

分局号 = 4{ 数字 }4

7.某公司为本科以上学历的人重新分配工作，分配原则如下：

- 1).如果年龄不满 18 岁，学历是本科，男性要求报考研究生，女性则担任行政工作；
- 2).如果年龄满 18 岁不满 50 岁，学历本科，不分男女，任中层领导职务，学历是硕士不分男女，任课题组组长；
- 3).如果年龄满 50 岁，学历本科，男性任科研人员，女性则担任资料员，学历是硕士不分男女，任课题组组长。

要求：画出得出判定表（化简）。

	1, 2, 3 7, 8, 9	4	5, 11	6	10	12
性 别	-----	M	-----	M	F	F
文化程度	G	U	U	U	U	U
年 龄	-----	L	Y	C	L	C
组长						
领导						
科研						
行政						
资料员						
考研						

8.某培训中心要研制一个计算机管理系统。它的业务是：

将学员发来的信件收集分类后，按几种不同的情况处理。

如果是报名的，则将报名数据送给负责报名事务的职员，他们将查阅课程文件，检查该课程是否额满，然后在学生文件、课程文件上登记，并开出报告单交财务部门，财务人员开出发票给学生。

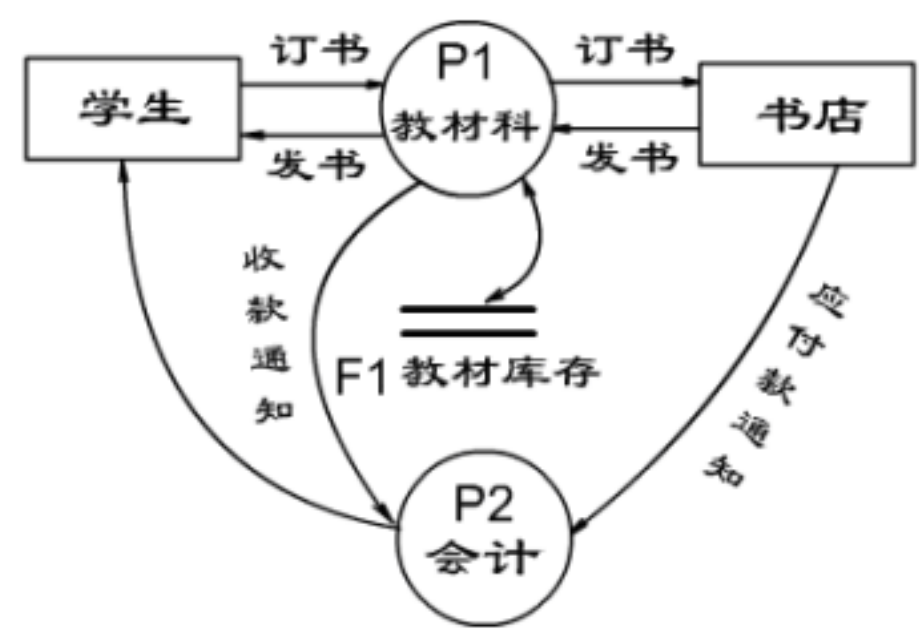
如果是想注销原来已选修的课程，则由注销人员在课程文件、学生文件和帐目文件上做相应的修改，并给学生注销单。

如果是付款的，则由财务人员在帐目文件上登记，也给学生一张收费收据。

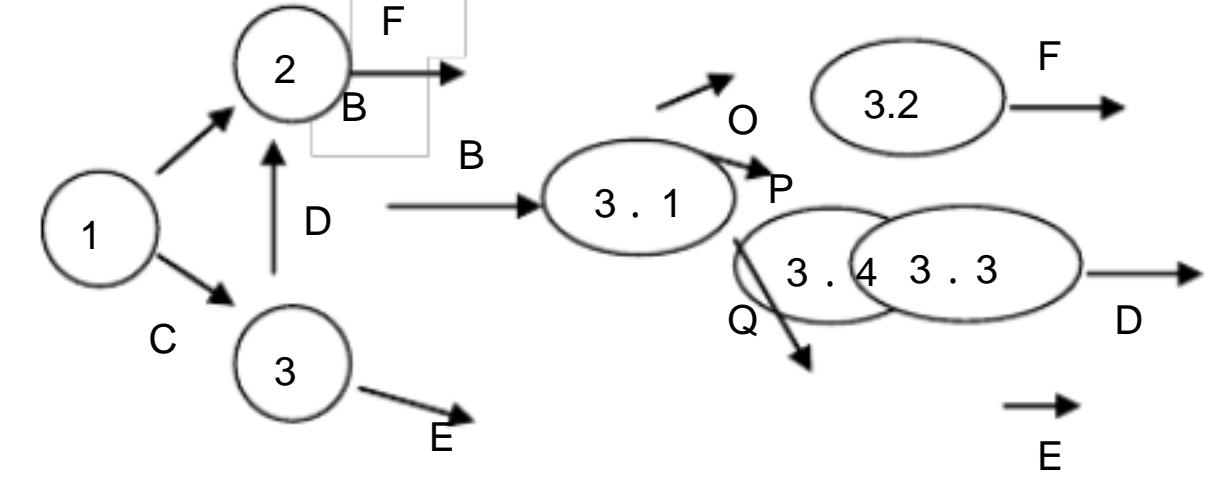
要求：对以上问题画出数据流程图。

9.根据下列描述，画出教材征订系统的第一层数据流图。学生入学后到教材科订书，教材科根据教材库存情况分析是否需要买书，如需购买，则向书店购买。各种资金往来通过学校的会计科办理。

答：



10. 指出下列数据流图中存在的问题。



父图

子图

参考答案：

- 1、父图没有输入数据流；
- 2、子图的输入数据流应该是 C，而不是 B；
- 3、子图的输出数据流应该只有 D 和 E，而不应有 F。

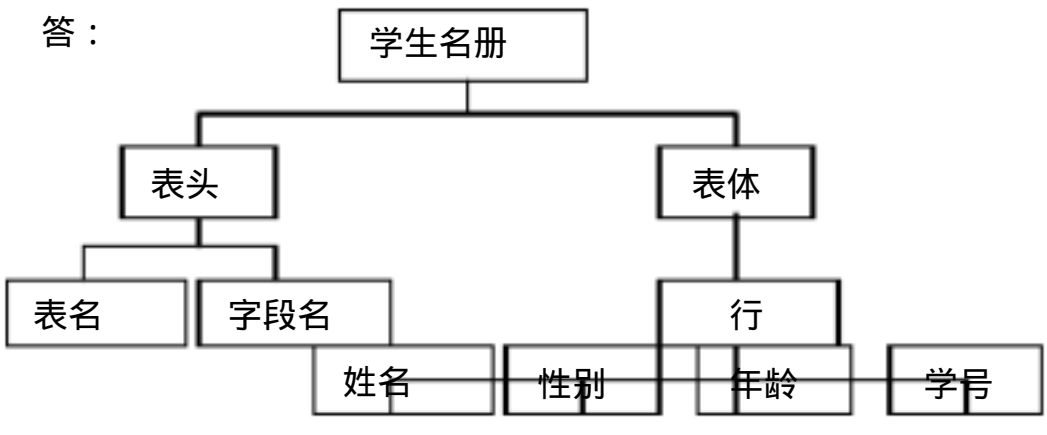
11. 用 Jackson 图表示下图所示的二维表格：

学生名册

姓名	性别	年龄	学号
.....

该学生名册由表头和表体两部分组成。其中表头又顺序包括表名和字段名。表体可由任意行组成，每行包括学生的姓名、性别、年龄和学号。

答：



(一) 名词解释

- 1. 软件概要设计
- 2. 模块
- 3. 模块化
- 4. 抽象

5. 信息隐蔽

6. 模块独立性

7. 耦合性

8. 无直接耦合

9. 数据耦合

10. 标记耦合

11. 控制耦合

12. 公共耦合

13. 内容耦合

14. 内聚性

15. 偶然内聚

16. 逻辑内聚

17. 时间内聚

18. 通信内聚

19. 顺序内聚

20. 功能内聚

21. 软件结构图

22. 结构化设计

23. 变换流

24. 事务流

（一）名词解释答案

1. 软件概要设计又称结构设计，这是一个把软件需求转换为软件表示（只是描述软件的总的体系结构）的过程。
2. 模块在程序中是数据说明、可执行语句等程序对象的集合，或者是单独命名和编址的元素，在软件的体系结构中，模块是可组合、分解和更换的单元。
3. 模块化指解决一个复杂问题时自顶向下逐层把软件系统划分成若干模块的过程。每个模块完成一个特定的子功能，所有模块按某种方法组装起来，成为一个整体，完成整个系统所要求的功能。
4. 抽象是认识复杂现象过程中使用的思维工具，即抽出事物本质的共同特性而暂不考虑它的细节，不考虑其他因素。
5. 信息隐蔽指在设计和确定模块时，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。
6. 模块独立性指每个模块只完成系统要求的独立的子功能，并且与其他模块的联系最少且接口简单。
7. 耦合性也称块间关系，指软件系统结构中各模块间相互联系紧密程度的一种度量。
8. 无直接耦合指两个模块之间没有直接关系，它们分别从属于不同模块的控制和调用，它们之间不传递任何信息。
9. 数据耦合指两个模块之间有调用关系，传递的是简单的数据值，相当于高级语言中值传递。
10. 标记耦合指两个模块之间传递的是数据结构。
11. 控制耦合指一个模块调用另一个模块时，传递的是控制变量（如开关、标志等），被调用模块通过该控制变量的值有选择地执行块内某一功能。
12. 公共耦合指通过一个公共数据环境相互作用的那些模块间的耦合。公共数据环境可以是全程变量或数据结构、共享的通信区、内存的公共覆盖区及任何存储介质上的文件、物理设备等。

13. 内容耦合指一个模块直接调用另一个模块的内部数据，或通过非正常入口而转入另一个模块内部。
14. 内聚性又称块内联系，指模块的功能强度的度量，即一个模块内部各个元素彼此结合的紧密程度的度量。
15. 偶然内聚指一个模块内的各处理元素之间没有任何联系。
16. 逻辑内聚指模块内执行几个逻辑上相似的功能，通过参数确定该模块完成哪一个功能。
17. 时间内聚指需要同时执行的动作组合在一起形成的模块。
18. 通信内聚指模块内所有处理元素都在同一个数据结构上操作（有时称为信息内聚），或者指各处理使用相同的输入数据或者产生相同的输出数据。
19. 顺序内聚指一个模块中各个处理元素都密切相关于同一功能且必须顺序执行，前一功能元素的输出就是下一功能元素的输入。
20. 功能内聚指模块内所有元素共同完成一个功能，缺一不可。因此模块不能再分割。
21. 软件结构图是软件系统的模块层次结构，反映了整个系统的功能实现。
22. 结构化设计又称面向数据流的设计，它是以需求分析阶段产生的数据流图为基础，按一定的步骤映射成软件结构。
23. 变换流指数据流图（简称 DFD）具有明显的输入、变换和输出界面。
24. 事务流指数据流图中输入流被分离成许多数据流，形成许多加工路径，并根据输入值选择其中一条路径来执行。

（一）名词解释

1. 详细设计
2. 结构化程序设计
3. 流程图

4. 盒图

5. 过程设计语言

6. JSP

7. JSD

(一) 名词解释答案

1. 详细设计主要确定每个模块的具体执行过程，也称过程设计。

2. 结构化程序设计是一种典型的面向数据流的软件总体设计方法。它采用自顶向下、逐步求精的设计方法和单入口单出口的控制结构，并且只包含顺序、选择和重复 3 种结构。

3. 流程图又称程序框图，是一种描述程序逻辑结构的工具。

4. 盒图（简称 N-S 图）。它是一种算法描述工具，它是一种由外往里、结构化逐层展开的二维盒型结构。盒图的控制流程为自上而下，从外到里地执行。

5. 过程设计语言（简称 PDL），也称程序描述语言，又称伪码，它是一种用于描述模块算法设计和处理细节的语言。

6. JSP 方法定义一组以数据结构为指导的映射过程，它根据输入、输出的数据结构，按一定的规则映射成软件的过程描述，即程序结构。适于详细设计阶段。

7. JSD 主要以活动事件为中心，通过由一串活动顺序组合构成进程，建立系统模型，最后实现该模型。

(二) 填空题

1. 软件结构的设计是以 _____ 为基础的。

2. 软件设计阶段，是把软件“ _____ ”的逻辑模型变换为“ _____ ”的物理模型。

3. 数据库的“概念设计”与“逻辑设计”分别对应于系统开发中的“_____”与“_____”，而数据库的“物理设计”与模块的“_____”相对应。

4. 概要设计文档主要有 _____、_____、_____和_____。

5. 模块具有 _____、_____、_____和_____几种基本属性。

6. 模块化指解决一个复杂问题时 _____逐层把软件系统划分成若干 _____的过程。

7. 衡量软件的独立性有两个定性的度量标准，即 _____和_____。

8. 模块之间联系越紧密，其 _____就越强，模块的 _____则越差。

9. 模块的耦合性由低到高分 _____、_____、_____、_____、_____、_____和_____。

10. 数据耦合传递的是 _____，标记耦合传递的是 _____，控制耦合传递的是 _____。

11. 软件结构往往用 _____结构的图形来表示。软件结构图是软件系统的 _____结构。

12. 面向数据流的设计又称 _____，它是以需求分析阶段产生的 _____为基础，按一定的步骤映射成软件结构。

13. 数据流图一般可分为 _____和_____两类。

(二) 填空题答案

1. 模块

2. 做什么 怎么做

3. 需求分析 概要设计 详细设计

4. 概要设计说明书 数据库设计说明书 用户手册 组装测试计划

5. 接口 功能 逻辑 状态

6. 自顶向下 模块

7. 耦合性 内聚性

8. 耦合性 独立性

9. 无直接耦合 数据耦合 标记耦合 控制耦合 外部耦合 公共耦合 内容耦合

10. 数据值 数据结构 控制变量

11. 树状或网状 模块层次

12. 结构化设计 数据流图

13. 变换型 事务型

(二) 填空题

1. 详细设计也称 _____, 它是软件设计的第二阶段, 主要确定每个 _____ 的具体执行过程。

2. 在详细设计阶段, 需要对处理过程的 _____ 和数据库的 _____ 结构做评审。

3. 处理过程设计中最典型的方法是 _____ 方法, 其基本要点是采用的程序设计方法是 _____、_____。

4. 任何程序都可由 _____、_____和_____ 3 种基本控制结构构造。这 3 种基本结构的共同点是 _____、_____。

5. 详细描述处理过程常用 3 种工具: _____、_____、和_____。

6. 程序流程图又称 _____, 它是历史最悠久、使用最广泛的一种描述程序 _____结构的工具。

7. 结构化流程图要求 3 种基本控制结构 _____和_____而成, 不能有相互交叉的情况。

8. PDL 是一种描述模块 _____设计和处理细节的语言, 其_____语法具有严格的关键字, _____语法使用自然语言的词汇。

9. PDL 是描述处理过程“ ”的细节，结构化语言是描述加工“ ”的。

10. Jackson 方法是面向 的设计方法，它的描述工具是 。

11. JSD 方法以 为中心，通过由一串活动顺序组合构成的 ，建立
模型，最后实现该模型。

12. JSP 方法定义了一组以 为指导的映射过程，它根据输入、输出的数据结构，按一定的规则映射成
软件的过程描述，即 ，而不是软件的 。

（二）填空题答案

1. 过程设计 模块

2. 算法 物理

3. 结构化程序设计 自顶向下 逐步求精

4. 顺序 选择 循环 单入口 单出口

5. 图形 表格 语言

6. 程序框图 逻辑

7. 顺序组合 完整嵌套

8. 算法 结构化

9. 怎么做 做什么

10. 数据结构 Jackson

11. 活动事件 进程 系统

12. 数据结构 程序结构 体系结构

(三) 单项选择题

1. 最高程度也是最差的耦合是 ()。

A. 公共耦合

B. 内容耦合

C. 控制耦合

D. 数据耦合
2. 概要设计阶段产生的文档不包括 ()。

A. 概要设计说明书

B. 数据库设计说明书

C. 用户手册

D. 开发进度月报
3. 一个模块把数值作为参数传送给另一个模块，这种耦合方式称为 ()。

A. 数据耦合

B. 公共耦合

C. 控制耦合

D. 标记耦合
4. 一个模块内部各程序段都在同一张表上操作，这个模块的内聚性称为 ()。

A. 时间内聚

B. 功能内聚

C. 通信内聚

D. 顺序内聚
5. 结构化设计是一种面向 () 的设计方法。

A. 数据流

B. 模块

C. 数据结构

D. 程序
6. 结构化设计又称为 ()。

- A. 概要设计
- B. 面向数据流设计
- C. 面向对象设计
- D. 详细设计

7. 反映模块内部特性的是（ ）。

- A. 接口
- B. 状态
- C. 功能
- D. 逻辑

8. 画软件结构图时应注意调用关系只能（ ）。

- A. 从上到下
- B. 从下到上
- C. 从左到右
- D. 从右到左

9. 变换型 DFD 是一个（ ）结构。

- A. 循环
- B. 选择
- C. 分层
- D. 顺序

（三）单项选择题答案

1.B 2.D 3.A 4.C 5.A 6.B 7.D 8.A 9.D

（三）单项选择题

1. 在详细设计阶段，经常采用的工具包括（ ）。

- A. SC 图
- B. DFD 图
- C. 判定表
- D. 形式化方法

2. 结构化程序设计的一种基本方法是（ ）。

- A. 筛选法
- B. 递归法

- C. 迭代法
- D. 逐步求法

3. 与详细设计相对应的是数据库的 ()设计。

- A. 概念
- B. 逻辑
- C. 物理
- D. 功能

4. 详细设计的基本任务中包括 ()。

- A. 数据流设计
- B. 人机对话设计
- C. 数据库的逻辑设计
- D. 软件结构设计

5. 下面说法不正确的是 ()。

- A. 流程图不易表示数据结构
- B. 流程图容易造成非结构化的程序结构
- C. 流程图支持逐步求精
- D. 流程图描述的是程序的逻辑结构

6. 下面说法不正确的是 ()。

- A. 盒图支持逐步求精
- B.盒图容易造成非结构化的程序结构
- C. 盒图描述的是算法
- D.盒图容易表达模块的层次结构

7. Jackson 方法以 ()为中心。

- A. 控制
- B. 数据结构
- C. 活动事件
- D. 数据流

8. Jackson 方法是面向 ()的设计方法。

- A. 数据结构
- B. 数据流
- C. 控制
- D. 对象

（三）单项选择题答案

1.C 2.D 3.C 4.B 5.C 6.B 7.B 8.A

（四）简答题

1. 什么是软件概要设计？该阶段的基本任务是什么？
2. 软件设计的基本原理包括哪些内容？
3. 什么是模块间的耦合性？有哪几种耦合性？简述降低模块间耦合度的方法。
4. 什么是软件结构？简述软件结构设计优化准则。
5. 变换分析设计与事务分析设计有什么区别？简述其设计步骤。

（四）简答题答案

1. 软件概要设计又称结构设计，这是一个把软件需求转换为软件表示（只是描述软件的总的体系结构）的过程。该阶段的基本任务是：

设计软件系统结构：

- 采用某种设计方法，将一个复杂的系统按功能划分成模块。
- 确定每个模块的功能。
- 确定模块之间的调用关系。
- 确定模块之间的接口，即模块之间传递的信息。
- 评价模块结构的质量。

数据结构和数据库设计：在概要设计阶段，数据结构设计应用抽象的数据类型、数据库设计应用数据库的逻辑设计。

编写概要设计文档：包括概要设计说明书、数据库设计说明书、用户手册、组装测试计划。

评审：对设计部分是否完整地实现了需求中规定的功能、性能等要求，设计方案的可行性，关键的处理及内外部接口定义正确性、有效性，各部分之间的一致性等一一进行评审。

2. 软件设计的基本原理包括：

模块化：指解决一个复杂问题时自顶向下逐层把软件系统划分成若干模块的过程。每个模块完成一个特定的子功能，所有模块按某种方法组装起来，成为一个整体，完成整个系统所要求的功能。模块化是软件解决复杂问题所具备的手段。

抽象：是认识复杂现象过程中使用的思维工具，即抽出事物本质的共同的特性而暂不考虑它的细节，不考虑其他因素。软件工程中每一步都是对软件解决方法的抽象层次的一次细化。

信息隐蔽：指在设计和确定模块时，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。通过信息隐蔽，可以定义和实施对模块的过程细节和局部数据结构的存取限制。

模块独立性。指每个模块只完成系统要求的独立的子功能，并且与其他模块的联系最少且接口简单。衡量模块独立性有两个标准耦合性和内聚性，可判断设计方案的优劣。

3. 模块间的耦合性是根据模块的外部特征提出的，是对软件系统结构中各模块间相互联系紧密程度的一种度量，也称块间关系。耦合程度由低到高分为 6 种：

无直接耦合（不传递任何消息）。

数据耦合（传递的是值）。

标记耦合（传递的是数据结构）。

控制耦合（传递的是控制变量，例如开关、标志等）。

外部耦合（传递的是 I/O 环境）

公共耦合（传递的是在公共数据环境中的数据）。

内容耦合（传递的是一个模块的内部数据，往往出现在汇编语言中）。

模块间联系越紧密，其耦合性越强，模块的独立性越差。提高模块独立性，降低模块间耦合度的措施是：

在耦合方式上降低模块间接口的复杂性。

- 模块接口方式不采用直接引用（内容耦合），而采用调用方式（例如过程语句调用）；
- 模块接口信息通过参数传递且传递信息的结构尽量简单，不用复杂数据结构（例如过程、指针等），参数个数不宜过多。

在传递信息类型上尽量使用数据耦合，避免控制耦合，慎用或有控制地使用公共耦合。

4. 软件结构设计优化准则如下：

划分模块时，尽量做到高内聚、低耦合，保持模块相对独立性，以此优化初始的软件结构。

一个模块的作用范围应在其控制范围之内，且判定所在的模块应与受其影响的模块在层次上尽量靠近。

软件结构的深度、宽度、扇入、扇出应适当。

模块的大小要适中。

模块的接口要简单、清晰、含义明确，便于理解，易于实现、测试与维护。

5. 变换分析设计是一个顺序结构，由输入、变换和输出三部分组成，其工作过程有 3 步：取得数据、变换数据和给出数据。事务分析设计是将它的输入流分离成许多发散的数据流，形成许多加工路径，并根据输入的值选择其中一个路径来执行。二者区别：变换分析设计适用于具有明显变换特征的数据流程图，事务分析设计适用于具有明显事务特征的数据流程图。

变换分析设计步骤：

确定 DFD 中的变换中心、逻辑输入和逻辑输出：

- 从物理输入端开始，沿着数据流方向向系统中心寻找，直到有这样的数据流，它不能再被看做是系统的输入，则它的前一个数据流是系统的逻辑输入。
- 从物理输出端开始，逆数据流方向向中间移动，可以确定系统的逻辑输出。
- 介于逻辑输入和逻辑输出之间的加工就是变换中心，即主加工。

设计软件结构的顶层和第一层——变换结构：

- 顶层即主模块，主要完成所有模块的控制。
- 第一层至少有 3 个功能模块：输入、输出和变换模块，即为逻辑输入设计一个输入模块，其功能是为顶层模块提供相应数据，为逻辑输出设计一个输出模块，其功能是输出顶层模块的信息，为变换中心设计一个变换模块，其功能是将逻辑输入进行变换加工，然后逻辑输出。

设计中、下层模块。对第一层的输入、变换、输出模块自顶向下逐层分解。

- 输入模块下属模块的设计：可设计两个下属模块，一个接收，一个转换。
- 输出模块下属模块的设计：可设计两个下属模块，一个转换，一个发送。
- 变换模块下属模块的设计：按照模块独立性原则来组织其结构，一般对每个基本加工建立一个功能模块。

设计的优化：

- 输入部分求精：为每个物理输入设置专门模块，其他输入模块与转换数据模块可适当合并。
- 输出部分求精：为每个物理输出设置专门模块，其他输出模块与转换数据模块可适当合并。
- 变换部分求精：根据设计准则，对模块进行合并或调整。

事务分析设计步骤：

确定 DFD 的事务中心和加工路径：当数据流图中每个加工将一个输入数据分解成多个发散的输出数据流时，该加工就是事务中心，从事务中心辐射出去的数据流为各个加工路径。

设计软件结构的顶层和第一层——事务结构。

顶层是一个全控模块，有两个部分：接收分支（与变换型 DFD 的输入部分设计方法相同）和发送分支（即调度模块，它控制管理所有下层的事务处理模块）。

设计中、下层模块、优化等工作，与变换分析设计相同。

（四）简答题

1. 什么是软件详细设计？该阶段的基本任务是什么？
2. 什么是结构化程序设计？简述结构化程序设计的基本要点。
3. 详细设计有哪几种描述方法？
4. Jackson 方法包括哪几步？运用 Jackson 方法进行软件设计时，在什么条件下可直接由数据结构导出软件结构？

（四）简答题答案

1. 详细设计是模块设计的第二阶段，主要确定每个模块的具体执行过程，也称过程设计。其基本任务有：

为每个模块进行详细的算法设计。用某种图形、表格、语言等工具将每个模块处理过程的详细算法描述出来。

为模块内的数据结构进行设计。

对数据库进行物理设计，即确定数据库的物理结构。

其他设计。根据软件系统类型，还可能要进行代码设计、输入/输出格式设计、人机对话设计。

编写详细设计说明书。

评审：评审处理过程的算法和数据库的物理结构。

2. 结构化程序设计是一种典型的面向数据流的软件总体设计方法，它采用采用自顶向下、逐步求精的设计方法和单入口单出口的控制结构，且只包含顺序、选择和重复 3 种结构。

基本要点是：

采用自顶向下、逐步求精的程序设计方法：详细设计中某个模块内部处理过程仍然可以逐步求精，降低处理细节的复杂程度。

使用 3 种基本控制结构（顺序、选择和重复，其共同点是单入口、单出口）构造程序：

- 用顺序方式对过程分解，确定各部分的执行顺序。
- 用选择方式对过程分解，确定某个部分的执行条件。
- 用循环方式对过程分解，确定某个部分进行循环的开始和结束的条件。
- 对于处理过程仍然模糊的部分反复使用以上分解方法，最终将所有细节确定下来。

主程序员组的组织形式：突出主程序员的领导，设计责任集中在少数人身上，有利于提高软件质量和软件生产率。其组织形式是：

- 一个主程序员：负责全部技术活动。
- 一个后备程序员：协调、支持主程序员。
- 一个程序管理员：负责事务性工作，如收集、记录数据，文档资料管理等。
- 一些专家（如通信专家、数据库专家）。
- 其他技术人员。

3. 详细描述处理过程常用 3 种工具：

图形：

- 结构化程序流程图是程序逻辑结构的描述工具。

- 盒图（N-S 图）描述常用的基本控制结构。

表格：

- 判定表是描述逻辑条件复杂的算法。

语言：

过程设计语言（简称 PDL）是一种用于描述模块算法设计和处理细节的语言。

4. Jackson 方法包括 5 步：

分析并确定输入数据和输出数据的逻辑结构，并用 Jackson 结构图表示这些数据结构。

找出输入数据结构和输出结构中有对应关系的数据单元：对应关系指数据单元在数据内容上、数量上和顺序上有直接的因果关系。

按一定的规则由输入、输出的数据结构导出程序结构。

列出基本操作与条件，并把它们分配到程序结构图的适当位置。

用伪码（也称图解逻辑）写出程序。

运用 Jackson 方法进行软件设计时，在数据结构与软件结构存在对应关系时，可直接由数据结构导出软件结构。

（五）应用题

1. 图 4-2 和图 4-3 是某公司员工工资管理的一部分，它们分别是同一功能的两个不同设计方案，你认为哪种设计方案较好？请陈述理由。

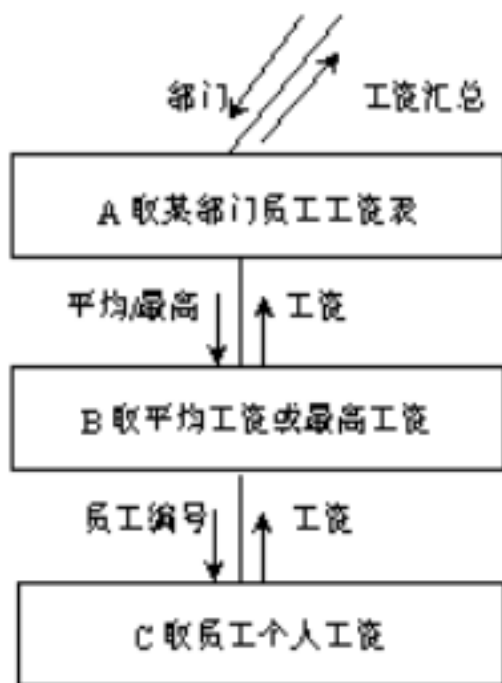


图 4-2 设计方案 1

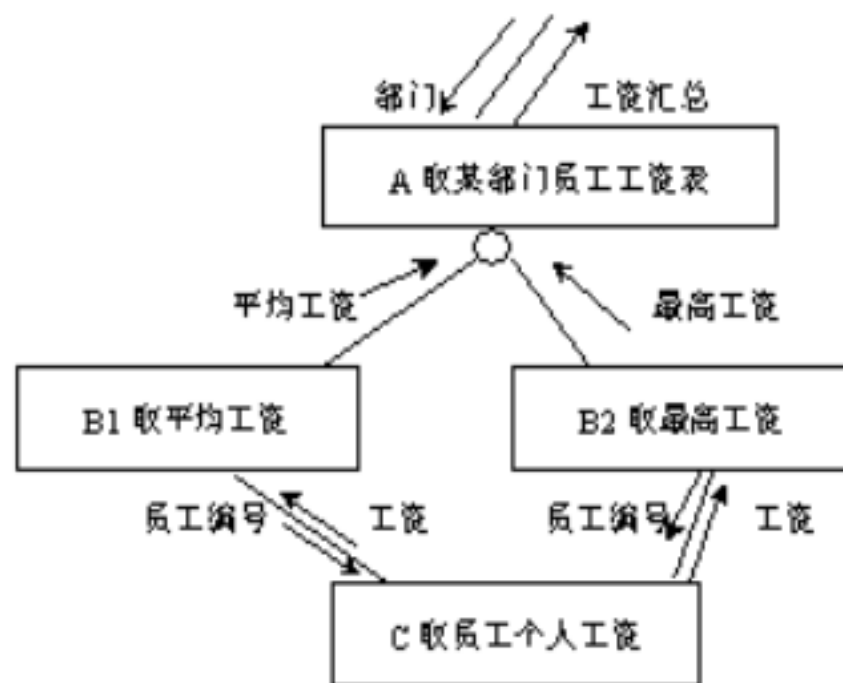


图 4-3 设计方案 2

2. 图 4-4 描述某银行安全监控系统的功能。请将该 DFD 图转换成软件结构图。



图 4-4 安全监控系统的 DFD图

(五) 应用题答案

1. 图 4-3 的设计方案好。 利用衡量模块独立性的两个标准耦合性和内聚性 （模块划分时应尽量做到高内聚、低耦合，提高模块独立性）判断设计方案优劣。图 4-2 中模块 B 的功能是取平均工资或最高工资，可见模块 B 包含两个功能，故模块 B 内聚弱，而模块 B 与模块 A 的耦合是控制耦合，耦合程度较高，模块的相对独立性差。 而在图 4-3 中，模块 B 分解成两个功能相对独立的模块 B1 和 B2，模块 B1 和 B2 内聚程度高，

模块 B1 和 B2 与模块 A 之间的耦合是数据耦合，耦合程度较低，因此模块的独立性好，所以图 4-3 设计方案好。

2.

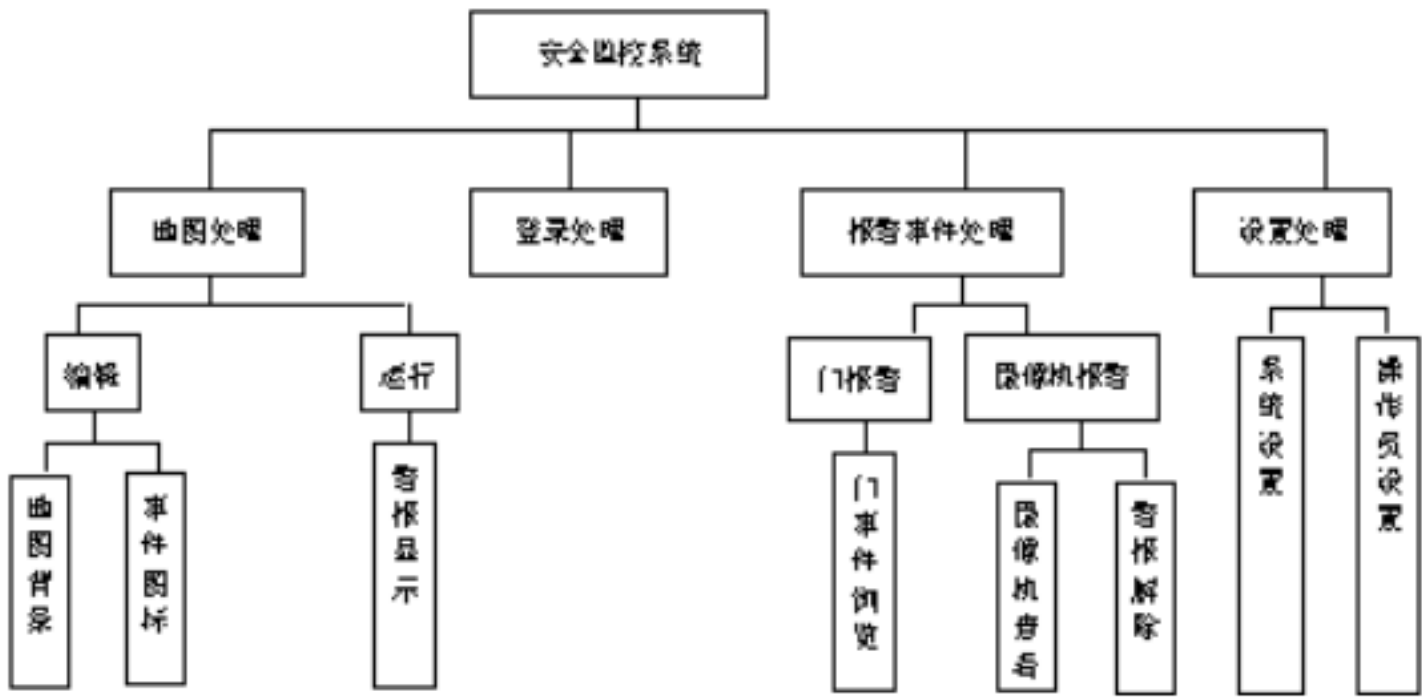


图 4-5 安全监控系统软件结构图

(五)应用题

1. 某商场在“五一”期间，顾客购物时收费有 4 种情况：普通顾客一次购物累计少于 100 元，按 A 类标准收费（不打折），一次购物累计多于或等于 100 元，按 B 类标准收费（打 9 折）；会员顾客一次购物累计少于 1000 元，按 C 类标准收费（打 8 折），一次购物累计等于或多于 1000 元，按 D 类标准收费（打 7 折）。请使用流程图、 PDL 来设计收费算法。

2. 一个正文文件由若干记录组成，每个记录是一个字符串，要求统计每个记录中空格字符的个数及文件中空格字符的总个数。要求输出数据格式是每复制一行字符串后，另起一行打印出这个字符串中的空格数，最后打印出文件空格的总个数，用 Jackson 方法设计该程序的程序结构。

(五)应用题答案 1.

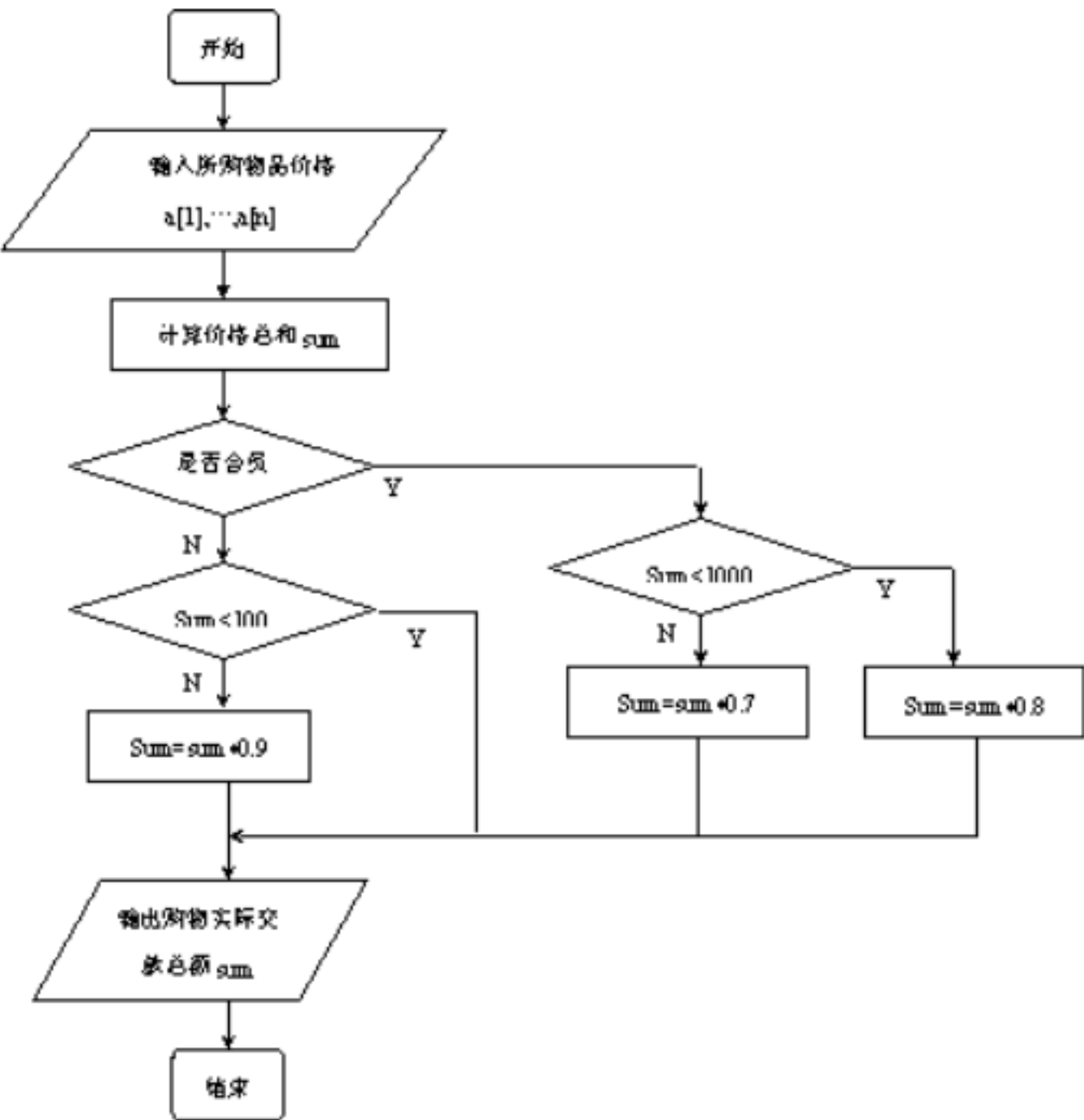


图 5-2 收费算法的流程图

收费算法的 PDL 描述：

```
COUNT_TOTAL (a[1],a[2], ... a[n])

{

    for(i=1; i<n; i++)    sum=sum+a[i];

    if ( 会员 )

    {

        if(sum<1000) sum=sum*0.8;

        else sum=sum*0.7;

    }

    else

    {

        if(sum<100) 不打折；

        else    sum=sum* 0.9;

    }

}
```

2. (1)建立输入输出数据结构，找出输入输出数据结构中的对应关系，如图 5-4 所示。

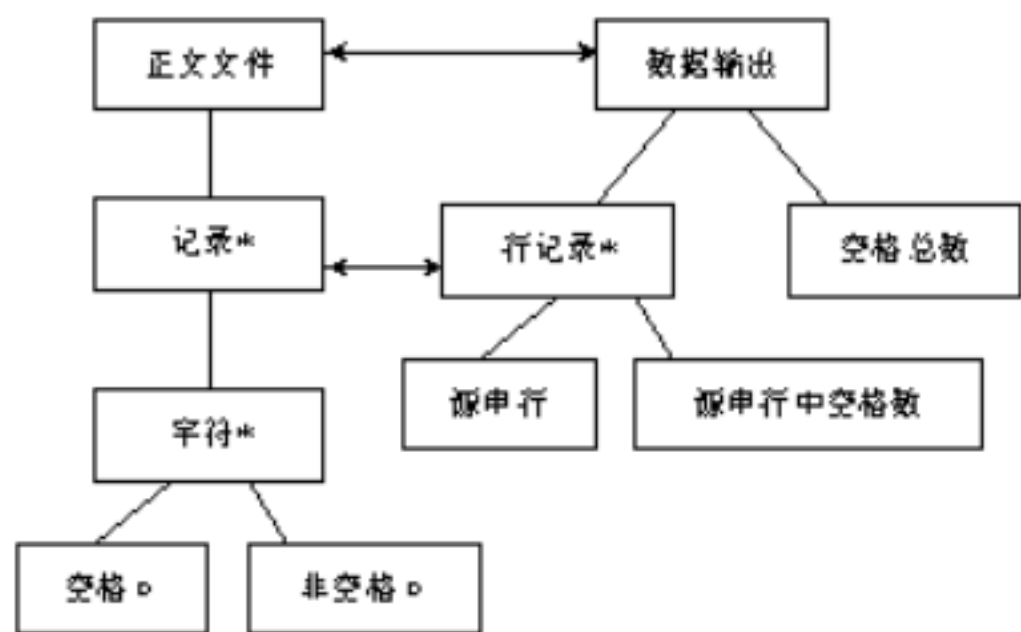


图 5-4 输入、输出数据结构中的对应关系

(2) 根据对应关系， Jackson图如图 5-5 所示。



图 5-5 Jackson 图

一、 选择题

1. 软件测试的目的是 ()。

A) 试验性运行软件
B) 发现软件错误
C) 证明软件正确
D) 找出软件中全部错误
2. 软件测试中白盒法是通过分析程序的 () 来设计测试用例的。

A) 应用范围
B) 内部逻辑
C) 功能
D) 输入数据
3. 黑盒法是根据程序的 () 来设计测试用例的。

A) 应用范围
B) 内部逻辑
C) 功能
D) 输入数据
4. 为了提高软件测试的效率，应该 ()。

A) 随机地选取测试数据
B) 取一切可能的输入数据作为测试数据

C) 在完成编码以后制定软件的测试计划
D) 选择发现错误可能性最大的数据作为测试用例

5. 与设计测试用例无关的文档是 ()。

- A) 项目开发计划 B) 需求规格说明书 C) 设计说明书 D) 源程序

6. 测试的关键问题是 ()。

- A) 如何组织软件评审
- B) 如何选择测试用例
- C) 如何验证程序的正确性
- D) 如何采用综合策略

7. 软件测试用例主要由输入数据和()两部分组成。

- A) 测试计划 B) 测试规则 C) 预期输出结果 D) 以往测试记录分析

8. 成功的测试是指运行测试用例后 ()。

- A) 未发现程序错误 B) 发现了程序错误 C) 证明程序正确性 D) 改正了程序错误

9. 下列几种逻辑覆盖标准中, 查错能力最强的是 ()。

- A) 语句覆盖 B) 判定覆盖 C) 条件覆盖 D) 条件组合覆盖

10. 在黑盒测试中, 着重检查输入条件组合的方法是 ()。

- A) 等价类划分法 B) 边界值分析法 C) 错误推测法 D) 因果图法

11. 单元测试主要针对模块的几个基本特征进行测试, 该阶段不能完成的测试是 ()。

- A) 系统功能 B) 局部数据结构 C) 重要的执行路径 D) 错误处理

12. 软件测试过程中的集成测试主要是为了发现()阶段的错误。

- A) 需求分析 B) 概要设计 C) 详细设计 D) 编码

13. 不属于白盒测试的技术是 ()。

- A) 路径覆盖 B) 判定覆盖 C) 循环覆盖 D) 边界值分析

14. 集成测试时,能较早发现高层模块接口错误的测试方法为()。

- A) 自顶向下渐增式测试 B) 自底向上渐增式测试 C) 非渐增式测试 D) 系统测试

15. 确认测试以 () 文档作为测试的基础。

- A) 需求规格说明书 B) 设计说明书 C) 源程序 D) 开发计划

16. 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。

- A) 程序内部逻辑 B) 程序的复杂度 C) 使用说明书 D) 程序的功能

17. 程序的三种基本结构是 ()。

- A) 过程子、程序、分程序 B) 顺序、选择、循环
- C) 递归、堆栈、队列 D) 调用、返回、转移

18. 结构化程序设计的一种基本方法是()

- A) 筛选法 B) 递归法 C) 归纳法 D) 逐步求精法

19. 软件调试的目的是 ()

- A) 找出错误所在并改正之
- B) 排除存在错误的可能性
- C) 对错误性质进行分类
- D) 统计出错的次数

20. 程序三种基本结构的共同特点是 ()

- A) 不能嵌套使用
- B) 只能用来写简单程序
- C) 已经用硬件实现
- D) 只有一个入口和一个出口

21. 下面说法正确的是 ()。

- A) 经过测试没有发现错误说明程序正确
- B) 测试的目标是为了证明程序没有错误
- C) 成功的测试是发现了迄今尚未发现的错误的测试
- D) 成功的测试是没有发现错误的测试

22. 覆盖准则最强的是 ()。

- A) 语句覆盖
- B) 判定覆盖
- C) 条件覆盖
- D) 路径覆盖

23. 发现错误能力最弱的是 ()。

- A) 语句覆盖
- B) 判定覆盖
- C) 条件覆盖
- D) 路径覆盖

24. 实际的逻辑覆盖测试中，一般以 () 为主设计测试用例。

- A) 条件覆盖
- B) 判定覆盖
- C) 条件组合覆盖
- D) 路径覆盖

25. 下面 () 方法能够有效地检测输入条件的各种组合可能引起的错误。

- A) 等价类划分
- B) 边界值分析
- C) 错误推测
- D) 因果图

26. 与确认测试阶段有关的文档是 ()。

- A) 需求规格说明书
- B) 概要设计说明书
- C) 详细设计说明书
- D) 源程序

27. 超出软件工程范围的测试是 ()。

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

28. 软件测试不需要了解软件设计的 ()。

- A) 功能
- B) 内部结构
- C) 处理过程
- D) 条件

29. ()方法需要考察模块间的接口和各模块之间的联系。

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

30. 调试应该由 ()完成。

- A) 与源程序无关的程序员
- B) 编制该源程序的程序员
- C) 不了解软件设计的机构
- D) 设计该软件的机构

31. 基线是软件生存期中各开发阶段的一个特定点，它可作为一个检查点，当采用的基线发生错误时，我们可以返回到最近和最恰当的 ()上。

- A) 配置项
- B) 程序
- C) 基线
- D) 过程

32. 版本管理是对系统不同版本进行的 ()过程

- A) 标识与跟踪
- B) 标识变更
- C) 发布变更
- D) 控制变更

33. 在变更控制中，管理各个用户存取和修改一个特定软件配置对象的权限是 ()。

- A) 异步控制
- B) 同步控制
- C) 存取控制
- D) 基线控制

34. 在配置管理中，“登入”和“检出”处理实现了两个重要的变更控制要素，即同步控制和 ()。

- A) 异步控制
- B) 同步控制
- C) 基线控制
- D) 存取控制

35. 按照软件配置管理的原始指导思想，受控制的对象应是 ()。

- A) 软件元素
- B) 软件配置项
- C) 软件项目
- D) 软件过程

36. 软件配置项是软件配置管理的对象，指的是软件工程过程中所产生的 ()。

- A) 接口
- B) 软件环境
- C) 信息项
- D) 版本

37. 在变更控制中，可用来确保由不同用户所执行的并发控制是 ()。

- A) 异步控制
- B) 同步控制
- C) 存取控制
- D) 基线控制

38. 以下说法错误的是 ()。

- A) 软件配置管理简称 SCI
- B) 软件配置项是配置管理的基本单位
- C) 软件配置实际上是一动态的概念
- D) 软件工程过程中某一阶段的变更均要引起软件配置的变更

39. 在设计人机界面时，应主要考虑的因素有 ()。

A) 系统响应时间 B) 错误处理 C) 用户求助机制 D) 以上都是

40. 人机界面的风格大致经过了()代的演变

A) 一 B) 二 C) 三 D) 四

二、 填空题

1. 软件测试的目的是尽可能多地发现软件中存在的 , 将 作为纠错的依据。

2. 测试阶段的基本任务是根据软件开发各阶段的 和程序的 , 精心设计一组 , 利用这些实例执行 , 找出软件中潜在的各种 和 。

3. 测试用例由 和预期的 两部分组成。

4. 软件测试方法一般分为两大类: 方法和 方法。

5. 动态测试通过 发现错误。根据 的设计方法不同, 动态测试又分为 与 两类。

6. 静态测试采用 和 的手段对程序进行检测。

7. 人工审查程序偏重于 的检验, 而软件审查除了审查 还要对各阶段 进行检验。

8. 计算机辅助静态分析利用 工具对测试程序进行 分析。

9. 黑盒法只在软件的 处进行测试, 依据 说明书, 检查程序是否满足 要求。

10. 白盒法必须考虑程序的 和 , 以检查 的细节为基础, 对程序中尽可能多的逻辑路径进行 。

11. 白盒测试是 测试, 被测对象是 , 以程序的 为基础设计测试用例。

12. 逻辑覆盖是对程序内部有 存在的逻辑结构设计测试用例, 根据程序内部的逻辑覆盖程度又可分为 、 、 、 、 和 6 种覆盖技术。

13. 实际的逻辑覆盖测试中, 一般以 覆盖为主设计测试用例, 然后再补充部分用例, 以达到 覆盖测试标准。

14. 循环覆盖是对程序内部有 存在的逻辑结构设计测试用例, 它通过限制 来测试。

15. 基本路径测试是在程序 基础上, 通过分析控制构造的 复杂性, 导出 集合, 从而设计测试用例。

16. 黑盒测试是 测试, 用黑盒技术设计测试用例有 4 种方法: 、 、 和 。

17. 等价类划分从程序的 说明, 找出一个输入条件(通常是 或), 然后将每个输入条件划分成两个或多个 。

18. 边界值分析是将测试 情况作为重点目标, 选取正好等于、刚刚大于或刚刚小于 的测试数据。如果输入或输出域是一个有序集合, 则应选取集合的 元素和 元素作为测试用例。

19. 在测试程序时, 根据经验或直觉推测程序中可能存在的各种错误, 称为 。

20．因果图的基本原理是通过画 _____图，把用自然语言描述的 _____转换为 _____，最后为 _____ 每一列设计一个测试用例。

21．测试的综合策略是在测试中，联合使用各种 _____方法。通常先用 _____法设计基本的测试用例，再用 _____法补充一些必要的测试用例。

22．软件测试过程中需要 3 类信息： _____、_____和_____。

23．软件测试一般经过 4 个测试： _____、_____、_____和_____。

24．_____指对源程序中每一个程序单元进行测试，检查各个模块是否正确实现规定的功能，_____ 从而发现模块在编码中或算法中的错误，它涉及 _____ 和_____的文档。

25．单元测试主要测试 _____的 5 个基本特征： _____、_____、_____、_____和_____。

26．在单元测试中， 需要为被测模块设计 _____ 模块和 _____ 模块。_____用来模拟被测模块的上级调用模块，用来代替被测模块所调用的模块。

27．集成测试指在 _____测试基础上，将所有模块按照设计要求组装成一个完整的系统进行的测试。也称 _____测试或 _____测试。

28．集成测试的方法有两种： _____和_____。

29．渐增式测试有两种不同的组装模块的方法： _____和_____。

30．自顶向下渐增式测试不需要编写 _____模块，只需要编写 _____模块，其步骤是从 _____模块开始，沿着被测程序的 _____的控制路径逐步向下测试，它有两种组合策略： _____和_____。

31．自底向上渐增式测试不需要编写 _____模块，只需要编写 _____模块。

32．确认测试指检查软件的 _____与_____是否与 _____说明书中确定的指标相符合，又称 _____测试。

33．确认测试在模拟环境下运用 _____测试方法，由 _____和_____参加的测试。

34．确认测试开始前需要制定 _____计划，结束后要写出 _____报告。其测试用例要选用 _____的数据。

35．软件配置审查的任务是检查软件的所有文档资料的 _____和_____。

36．调试也称 _____，是在成功的测试之后才开始进行，其目的是确定错误的 _____和_____，并改正错误。

37．调试技术包括 _____、_____、_____和_____。

38．回溯法调试是从 _____出发，而归纳法调试是从 _____入手。

39．被测试程序不在机器上运行，而是采用人工检测和计算机辅助分析检测的手段称为 _____测试。

40．用等价类划分法设计一个测试用例时，使其覆盖 _____尚未被覆盖的合理等价类。

41．用等价类划分法设计一个测试用例时，使其覆盖 _____不合理等价类。

42．在单元测试时，需要为被测模块设计 _____。

- 43．在集成测试时有两种测试方法，它们是_____。
- 44．软件测试是为了_____而执行程序的过程。
- 45．运行被测程序的方法称为_____测试。
- 46．动态测试中，主要测试软件功能的方法称为_____法。
- 47．选择测试用例，使得被测程序中每个判定的每个分支至少执行一次，_____这种逻辑覆盖标准称为_____。
- 48．要覆盖含循环结构的所有路径是不可能的，一般通过限制_____来测试。
- 49．用等价类划分法设计测试用例时，如果被测程序的某个输入条件规定了取值范围，则可确定一个合理的等在和_____。
- 50．凭经验或直觉推测程序中可能存在的错误而设计测试用例的方法是_____。
- 51．集成测试中的具体方法是_____。
- 52．确认测试阶段的两项工作是_____。
- 53．在单元测试中，测试一个模块时，需要设计_____。
- 54．软件配置管理，简称 SCM，它用于整个软件工程过程。其主要目标是：_____、_____、_____、_____。
- 55．SCM 是一组管理整个软件生存期各阶段中_____的活动。
- 56．基线的作用是把各阶段的开发工作划分得更加明确，便于检查与确认阶段成果。因此，基线可以作为项目的一个_____。

三、 名词解释

- 1．软件测试
- 2．静态测试
- 3．动态测试
- 4．黑盒测试
- 5．白盒测试
- 6．语句覆盖
- 7．判定覆盖
- 8．条件覆盖
- 9．判定 /条件覆盖
- 10．条件组合覆盖
- 11．路径覆盖
- 12．测试用例

13．驱动模块

14．桩模块

15．单元测试

16．集成测试

17．确认测试

18．渐增式测试

19．非渐增式测试

20．调试

21．人的因素的含义

22．基线

23．软件配置管理

24．软件配置项

四、 简答题

1．为什么说软件测试是软件开发中不可缺少的重要一环，但不是软件质量保证的安全网？

2．软件测试的目的是什么？为什么把软件测试的目的定义为只是发现错误？

3．软件测试应当遵循什么原则？为什么要遵循这些原则？

4．软件测试的步骤是什么？这些测试与软件开发各阶段之间的关系？

5．软件测试的过程是什么？

6．单元测试、集成测试和确认测试各自主要目标是什么？它们之间有什么不同？相互有什么关系？

7．什么是黑盒测试与白盒测试？它们都适应哪些测试？

8．简述软件测试与软件调试的区别。

9．软件配置管理的任务。

10．试述第三代界面的优点。

11．试述人机界面的设计过程。

五、 综合题

1．变量的命名规则一般规定如下：变量名的长度不多于 40 个字符，第一个字符必须为英文字母，其他字母可以英文字母、数字以及下划线的任意组合。请用等价分类法设计测试用例。

2．某商场在“五一”期间，顾客购物时收费有 4 种情况：普通顾客一次购物累计少于 100 元，按 A 类标准收费（不打折），一次购物累计多于或等于 100 元，按 B 类标准收费（打 9 折）；会员顾客一次购物累计少于 1000 元，按 C 类标准收费（打 8 折），一次购物累计等于或多于 1000 元，按 D 类标准收费（打 7 折）。

测试对象是按以上要求计算顾客收费模块，按照路径覆盖法设计测试用例。

3．下面是某程序的流程图：

（1）计算它的环路复杂性。

（2）为了完成基本路径测试，求它的一组独立的路径。

六、 选择题

- 1 . B 2 . B 3 . C 4 . D 5 . A 6 . B 7 . C 8 . B 9 . D 10 . D
11 . A 12 . B 13 . D 14 . A 15 . A 16 . A 17 . B 18 . D 19 . A 20 . D
21 . C 22 . D 23 . A 24 . C 25 . D 26 . A 27 . D 28 . D 29 . B 30 . B
31 . C 32 . A 33 . D 34 . B 35 . C 36 . B 37 . B 38 . A 39 . D 40 . D

七、 填空题

1. 错误 测试结果
2. 文档资料 内部结构 测试用例 程序 错误 缺陷
3. 输入数据 输出数据
4. 动态测试 静态测试
5. 运行程序 测试用例 黑盒测试 白盒测试
6. 人工检测 计算机辅助静态分析
7. 编码质量 编码 软件产品
8. 静态分析 特性
9. 接口 需求规格 功能
10. 内部结构 处理过程 处理过程 测试
11. 结构 源程序 内部逻辑
12. 判定 语句覆盖 判定覆盖 条件覆盖 判定 /条件覆盖 条件组合覆盖 路径覆盖
13. 条件组合 路径
14. 循环 循环次数
15. 控制流程图 环路 基本路径
16. 功能 等价类划分 边界值分析 错误推测 因果图
17. 功能 一句话 一个短语 等价类
18. 边界 边界值 第一个 最后一个
19. 错误推测法
20. 因果 功能说明 判定表 判定表
21. 测试 黑盒 白盒
22. 软件配置 测试配置 测试工具
23. 单元测试 集成测试 确认测试 系统测试
24. 单元测试 编码 详细设计
25. 模块 模块接口 局部数据结构 重要的执行路径 错误处理 边界条件
26. 驱动 桩 驱动模块 桩模块
27. 单元 组装 联合
28. 非渐增式测试 渐增式测试
29. 自顶向下结合 自底向上结合
30. 驱动 桩 顶层 软件结构图 深度优先策略 宽度优先策略
31. 桩 驱动

- 32. 功能 性能 需求规格 有效性
- 33. 黑盒 专门测试人员 用户
- 34. 测试 测试分析 实际运用
- 35. 完整性 正确性
- 36. 纠错 原因 位置
- 37. 简单调试 归纳法调试 演绎法调试 回溯法调试
- 38. 程序产生错误的地方 测试结果发现的线索
- 39. 静态
- 40. 尽可能多的
- 41. 一个
- 42. 驱动模块与桩模块
- 43. 渐增式和非渐增式
- 44. 发现错误
- 45. 动态测试
- 46. 黑盒法
- 47. 判定覆盖
- 48. 循环次数
- 49. 两个不合理的等价类
- 50. 错误推测法
- 51. 渐增式和非渐增式测试方法
- 52. 进行确认测试和软件配置审查
- 53. 驱动模块和桩模块
- 54. 标识变更 控制变更 确保变更正确地实现 报告有关变更
- 55. 变更
- 56. 检查点

八、 名词解释

1. 软件测试指为了发现软件中的错误而执行软件的过程。它的目标是尽可能多地发现软件中存在的错误，将测试结果作为纠错的依据。
2. 静态测试指被测试的程序不在机器上运行，而是采用人工检测和计算机辅助静态分析的手段对程序进行检测。
3. 动态测试指通过运行程序发现错误。
4. 黑盒测试指把测试对象看成一个黑盒子，测试人员完全不考虑程序的内部结构和处理过程，只在软件的接口处进行测试，依据需求规格说明书，检查程序是否满足功能要求，又称为功能测试或数据驱动测试。
5. 白盒测试指把测试对象看成一个打开的盒子，测试人员需了解程序的内部结构和处理过程，以检查处理过程的细节为基础，对程序中尽可能多的逻辑路径进行测试，检验内部控制结构和数据结构是否有错，实际的运行状态与预期的状态是否一致。
6. 语句覆盖指设计足够的测试用例，使被测程序中每个语句至少执行一次。
7. 判定覆盖指设计足够的测试用例，使被测程序中每个判定表达式至少获得一次“真”值或“假”值，从而使程序的每个分支至少都通过一次，因此判定覆盖又称分支覆盖。
8. 条件覆盖指设计足够测试用例，使判定表达式中每个条件的各种可能的值至少出现一次。
9. 判定 /条件覆盖指设计足够的测试用例， 使得判定表达式中每个条件的所有可能取值至少出现一次， 并使每个判定表达式所有可能的结果也至少出现一次。
10. 条件组合覆盖指设计足够的测试用例，使得每个判定表达式中条件的各种可能的值的组合都至少出现一次。

11. 路径覆盖指设计足够的测试用例，覆盖被测程序中所有可能的路径。
12. 测试用例指为寻找程序中的错误而精心设计的一组测试数据。
13. 驱动模块指用来模拟被测模块的上级调用模块，其功能比真正的上级模块简单得多，它只完成接受测试数据，以上级模块调用被测模块的格式驱动被测模块，接收被测模块的测试结果并输出。
14. 桩模块指用来代替被测试模块所调用的模块，其作用是返回被测试模块所需的信息。
15. 单元测试指对源程序中每一个程序单元进行测试，检查各个模块是否正确实现规定的功能，从而发现模块在编码中或算法中的错误。
16. 集成测试指在单元测试基础上，将所有模块按照设计要求组装成一个完整的系统进行的测试。也称组装测试或联合测试。
17. 确认测试指检查软件的功能与性能是否与需求规格说明书中确定的指标相符合，又称有效性测试。
18. 渐增式测试指逐个把未经过测试的模块组装到已经过测试的模块上去，进行集成测试。每加入一个新模块进行一次集成测试，重复此过程直到程序组装完毕。
19. 非渐增式测试指首先对每个模块分别进行单元测试，然后把所有的模块按设计要求组装在一起进行测试。
20. 调试指确定错误的原因和位置，并改正错误的过程，也称纠错。
21. 1) 人对感知过程的认识，包括视觉、阅读时的认知心理、记忆、归纳与演绎推理等；
2) 用户已有的技能和行为方式；
3) 用户所要求的完成的整个任务以及用户对人机交互部分的特殊要求。
22. 已经通过正式复审和批准的某规约或产品，它因此可以作为进一步开发的基础，并且只能遵循正式的变化控制过程得到改变。
23. 软件配置管理，简称 SCM，它用于整个软件工程过程。其主要目标是：标识变更、控制变更、确保变更正确地实现、报告有关变更。SCM 是一组管理整个软件生存期各阶段中变更的活动。
24. 软件配置项是软件工程中产生的信息项，它是配置管理的基本单位，对已成为基线的 SCI，虽然可以修改，但必须按照一个特殊的正确的过程进行评估，确认每一处的修改。

九、简答题

1. 软件测试是软件开发中不可缺少的重要一环，原因是：
 - 测试的工作量约占整个项目开发工作量的 40%左右，几乎一半。如果是关系到人的生命安全的软件，测试的工作量还要成倍增加。
 - 软件测试代表了需求分析、设计、编码的最终复审。
软件测试不是软件质量保证的安全网，因为软件测试只能发现错误，不能保证没有错误。
2. 软件测试的目的有：
 - 软件测试是为了发现错误而执行程序的过程。
 - 一个好的测试用例能够发现至今尚未发现的错误。
 - 一个成功的测试是发现了至今尚未发现的错误。

软件测试的目标定义为只是发现错误，原因是软件测试可以有两个目标，一个是预防错误，另一个是发现错误。由于软件开发是人的创造性劳动，人的活动不可能完美无缺，错误可能发生在任何一个阶段，因此预防错误这一目标几乎是不可实现的，所以软件测试的目标定义为只是发现错误。
3. 软件测试应当遵循原则如下：
 - 用例由输入数据和预期的输出数据两部分组成，因为这样便于对照检查，做到有的放矢。
 - 用例不仅选用合理的输入数据，还要选择不合理的输入数据。因为当以特殊方式使用程序时，会突然发现程序中有许多错误，故使用预期的不合理的输入数据进行程序测试，比用合理的输入数据收获要大，从而能更多地发现错误，提高程序可靠性。
 - 除了检查程序是否做了它应该做的事，还应该检查程序是否做了它不应该做的事，因为如果程序做了它不应该做的事，即使程序能做它应该做的事，程序也是错误的。

应制定测试计划并严格执行，因为这样可以排除随意性。

长期保留测试用例，因为测试用例的设计耗费很大的工作量，而修改后的程序可能有新的错误，需要进行回归测试，故必须将测试用例作为文档保存，使测试具有可重复性，同时测试用例是将来系统维护测试与确认的依据，保存测试用例也为以后的维护提供方便。

对发现错误较多的程序段，应进行更深入的测试，因为发现错误较多的程序段，其质量较差，同时在修改错误过程中又容易引入新的错误。

程序员避免测试自己设计的程序，因为测试目的是找错。从心理学角度讲，程序员大多对自己的程序存有偏见，总认为没有错误或错误不大，另外程序员对需求规格说明的理解而引入的错误则更难发现，应该由别人或另外的机构来测试会更客观、更有效。

4. 软件测试的步骤如图 7-3 所示。

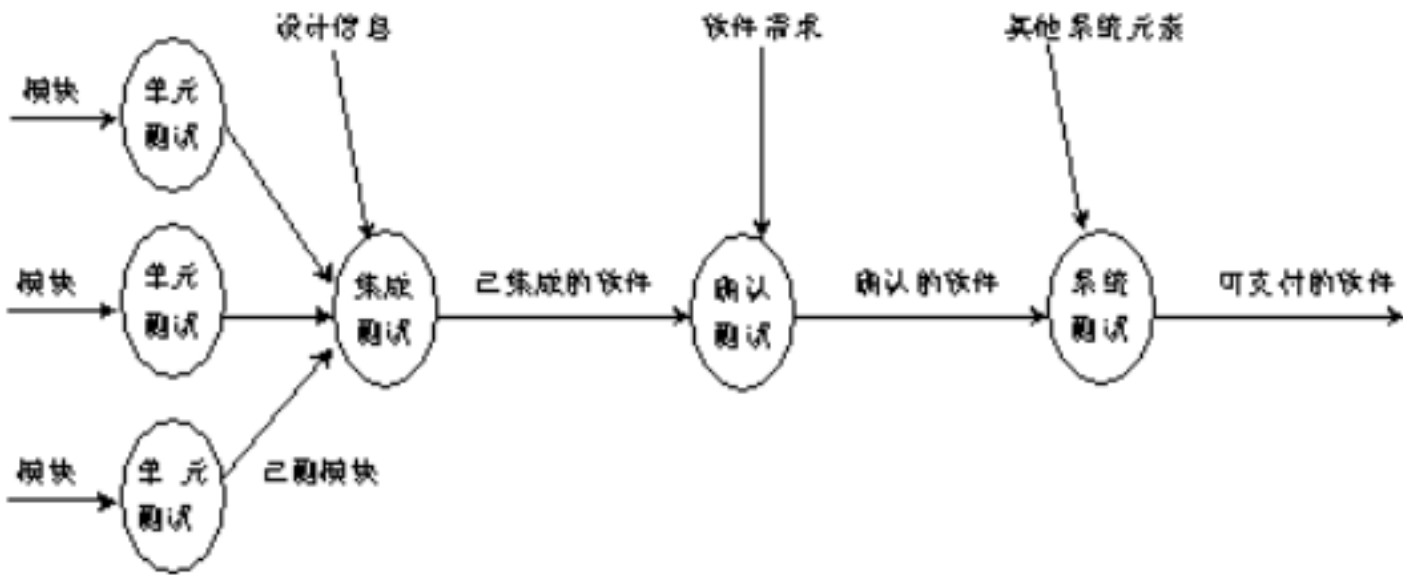


图 7-3 软件测试的步骤

这些测试与软件开发各阶段之间的关系如图 7-2 所示。因为系统测试已经超出了软件工程的范围，所以图 7-2 中的系统测试不包括在内。

5. 软件测试是一个规则的过程，包括测试设计、测试执行以及测试结果比较等。

测试设计：根据软件开发各阶段的文档资料和程序的内部结构，利用各种设计测试用例技术精心设计测试用例。

测试执行：利用这些测试用例执行程序，得到测试结果。

测试结果比较：将预期的结果与实际测试结果进行比较，如果二者不符合，对于出现的错误进行纠错，并修改相应文档。修改后的程序还要进行再次测试，直到满意为止。如果测试发现不了错误，可能由于测试配置考虑不周到，应考虑重新制定测试方案，设计测试用例。

6. 单元测试、集成测试和确认测试各自主要目标是：

单元测试的主要目标是检查各个模块是否正确实现规定的功能，从而发现模块在编码中或算法中的错误。

集成测试的主要目标是检查与设计相关的软件体系结构的有关问题。

确认测试的主要目标是检查已实现的软件是否满足需求规格说明书中确定的各种需求。

单元测试、集成测试和确认测试之间的不同如下：

测试内容不同：单元测试集中于单个模块的功能和结构检验，其测试内容主要包括模块接口、局部数据结构、重要的执行路径、错误处理和边界测试；集成测试集中于模块组合的功能和软件结构检验，其测试内容主要包括模块组装中可能出现的问题，即数据穿过接口可能丢失、一个模块可能破坏另一个模块的内容、子功能组装可能不等于主功能、全程数据结构问题、误差累积问题；确认测试集中于论证软件需求的可追溯性，主要包括测试软件功能和性能是否与软件需求一致、测试软件配置的所有程序与文档是否正确完整而且一致。

测试的方法不同：单元测试总是使用白盒测试法，为被测模块设计驱动模块和桩模块；集成测试使用渐增式测试和非渐增式测试，渐增式测试又有分为自顶向下结合法和自底向上结合法；确认测试总是使用黑盒测试法。

发现的错误不同：单元测试发现的错误主要是在编码阶段产生的错误，集成测试发现的错误主要是在设计阶段产生的错误，确认测试发现的错误主要是在需求分析阶段产生的错误。

涉及的文档不同：单元测试涉及编码和详细设计文档，集成测试涉及详细设计文档和概要设计文档，确认测试涉及软件需求规格说明书和用户手册。

三者相互关系是：单元测试、集成测试和确认测试是顺序实现的。首先单元测试对各个模块进行测试，然后集成测试以单元测试为基础，将所有已测模块按照设计要求组装成一个完整的系统，对模块组合的功能和软件结构检验进行测试，最后确认测试是以集成测试为基础，测试集成的软件是否满足需求规格说明书中确定的各种需求。

7. 黑盒测试指把测试对象看成一个黑盒子，测试人员完全不考虑程序的内部结构和处理过程，只在软件的接口处进行测试，依据需求规格说明书，检查程序是否满足功能要求，又称为功能测试或数据驱动测试。

白盒测试指把测试对象看成一个打开的盒子，测试人员需了解程序的内部结构和处理过程，以检查处理过程的细节为基础，对程序中尽可能多的逻辑路径进行测试，检验内部控制结构和数据结构是否有错，实际的运行状态与预期的状态是否一致。

白盒测试适应的测试有单元测试、逻辑覆盖（按逻辑覆盖程度不同，有语句覆盖、判定覆盖、条件覆盖、判定/条件覆盖、条件组合覆盖和路径覆盖。）循环覆盖（限制循环次数，有单循环和嵌套循环）和基本路径测试。

黑盒测试适应的测试有：确认测试、等价类划分、边界值分析、错误推测和因果图。

8. 软件测试与软件调试在目的、技术和方法等方面存在很大的区别，主要表现在：

测试从一个侧面证明程序员的失败，而调试是为了证明程序员的正确。

测试从已知条件开始，使用预先定义的程序，且有预知的结果，不可预见的只是程序是否通过测试。调试一般以不可知的内部条件开始，除统计性调试外，结果是不可预见的。

测试是有计划的，并要进行测试设计，而调试是不受时间约束的。

测试是一个发现错误、改正错误、重新测试的过程，而调试是一个推理过程。

测试的执行是有规程的，而调试的执行往往要求程序员进行必要的推理及知觉的飞跃。

测试经常由独立的测试组在不了解软件设计的前提下完成，而调试必须由了解详细设计的程序员完成。

大多数测试的执行和设计可由工具支持，而调试时，程序员能利用的工具主要是调试器。

十、综合题

1. 划分等价类下表。

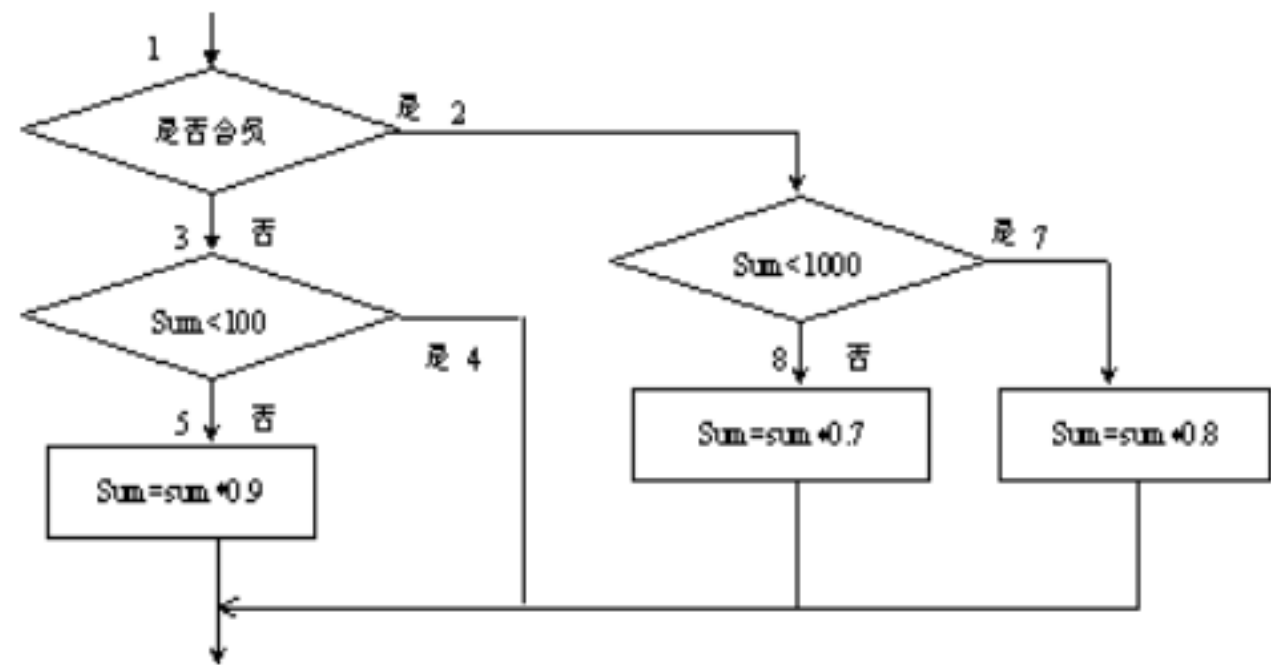
等价类划分表		
输入条件	合理等价类	不合理等价类
长度	(1)小于 40 个字符 (2)等于 40 个字符	(5)大于 40 个字符
第 一 个 字 符	(3)英文字母	(6)非英文字母
其他字母	(4)英文字母、数字或下划线的任意组合	(7)空格 (8)标点符号 (9)运算符号 (10)其他可显示字符

设计测试用例下表。

测试用例表		
测试数据	测试范围	期望结果
(1) s_name12	等价类 (1),(3),(4)	有效
(2) a1b2c3d4e5f6g7h8i9j1k2l3m4n5o6p7q8r9s_tr	等价类 (2),(3),(4)	有效

(3) a1b2c3d4e5f6g7h8i9j1k2l3m4n5o6p7q8r9s_trff	等价类 (5)	无效
(4) 234name	等价类 (6)	无效
(5) ab gh	等价类 (7)	无效
(6) ab!2f	等价类 (8)	无效
(7) fg+ghh	等价类 (9)	无效
(8) H@gh	等价类 (10)	无效

2. 被测模块的程序流程图如图所示。



被测模块的程序流程图

按照路径覆盖法设计测试用例如下：

- 是会员，累计消费 900 元，覆盖路径 127
- 是会员，累计消费 2000 元，覆盖路径 128
- 不是会员，累计消费 80 元，覆盖路径 134
- 不是会员，累计消费 300 元，覆盖路径 135

3. (a) 环路复杂性 = 判断数 + 1 = 6 + 1 = 7 (个)

(b) 路径 1：0 — — 13 — 19

路径 2：0 — — — 14 — 19

路径 3：0 — — — — 15 — 19

路径 4：0 — — — — — 16 — 19

路径 5：0 — — — — — — 17 — 19

路径 6：0 — — — — — — 18 — 19

路径 7：0 — — — — — — 12 — 19

4. (1) 划分等价类见表 1。

表 1 等价类划分表

输入条件	合理等价类	不合理等价类
长度	(1) 小于 40 个字符	(5) 大于 40 个字符
	(2) 等于 40 个字符	
第一个字符	(3) 英文字母	(6) 非英文字母
其他字母	(4) 英文字母、数字或下划线的任意组合	(7) 空格 (8) 标点符号 (9) 运算符
		(10) 其他可显示字符

(2) 设计测试用例见表 2。

表 2 测试用例表

测试数据	测试范围	期望结果
(1)s_name12	等价类 (1),(3),(4)	有效
(2)a1b2c3d4e5f6g7h8i9j1k2l3m4n5o6p7q8r9s_tr	等价类 (2),(3),(4)	有效
(3)a1b2c3d4e5f6g7h8i9j1k2l3m4n5o6p7q8r9s_trff	等价类 (5)	无效
(4)234name	等价类 (6)	无效
(5)ab gh	等价类 (7)	无效
(6)ab!2f	等价类 (8)	无效
(7)fg+ghh	等价类 (9)	无效
(8)H@gh	等价类 (10)	无效

5 . 等价划分法

划分等价类并编号，如下表所示。

十六进制整型常量输入条件的等价类表

输入数据	合理等价类	不合理等价类
十六进制整数	1. 0x 或 0X 开头 1 ~ 2 位数字串 2. 以 - 0x 打头的 1 ~ 2 位数字串	3. 非 0x 或非 - 打头的串 4. 含有非数字且 (a,b,c,d,e,f) 以外字符 5. 多于 5 个字符 6. - 后跟非 0 的多位串 7. - 0 后跟数字串 8. - 后多于 3 个数字
十六进制数范围	9. 在 - 7f ~ 7f 之间	10. 小于 - 7f 11. 大于 7f

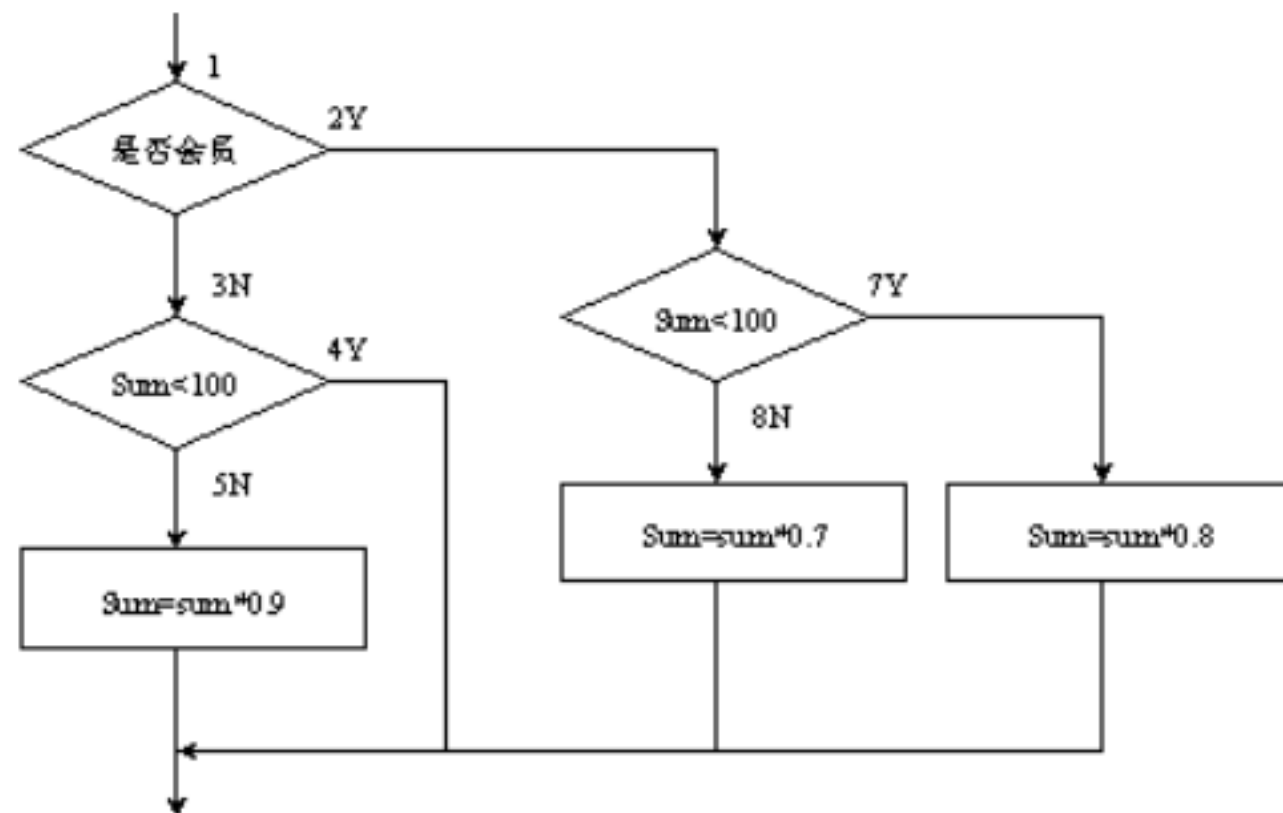
为合理等价类设计测试用例，表中有三个合理等价类，两个例子。

测试数据	期望结果	覆盖范围
0 x 23	显示有效输入	1 , 9
- 0 x 15	显示有效输入	2 , 9

为每个不合理等价类至少设计一个测试用例。

测试数据	期望结果	覆盖范围
2	显示无效输入	3
G12	显示无效输入	4
123311	显示无效输入	5
- 1012	显示无效输入	6
- 011	显示无效输入	7
- 0134	显示无效输入	8
- 0x777	显示无效输入	10
0x87	显示无效输入	11

6 . (1) 被测模块的程序流程图如下图所示。



(2) 按照路径覆盖法设计测试用例如下：

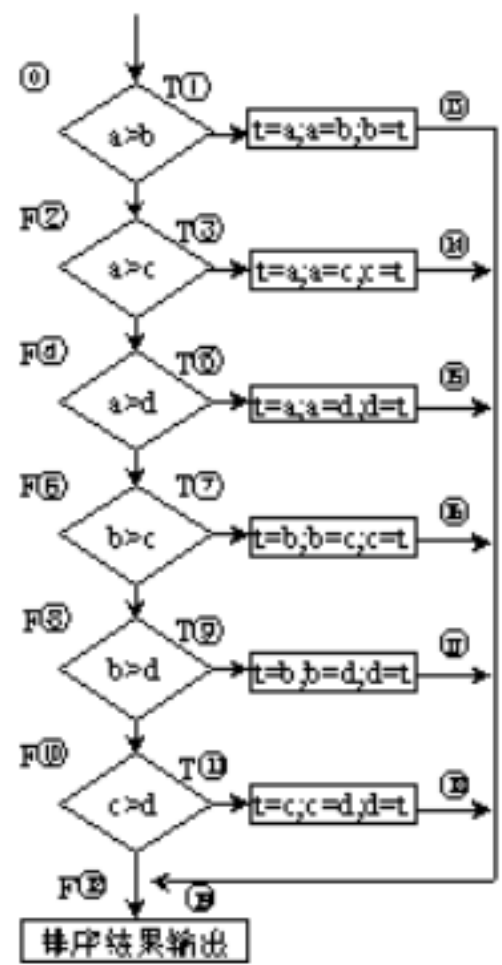
是会员，累计消费 900 元，覆盖路径 127

是会员，累计消费 2000 元，覆盖路径 128

不是会员，累计消费 80 元，覆盖路径 134

不是会员，累计消费 300 元，覆盖路径 135

2 .



4 . 变量的命名规则一般规定如下：变量名的长度不多于 40 个字符，第一个字符必须为英文字母，其他字母可以是英文字母、数字以及下划线的任意组合。请用等价分类法设计测试用例。

5 . 根据下列条件使用等价划分法设计测试用例。

某 8 位微机，其十六进制常数定义为：以 0x 或 0X 开头的数是十六进制整数，其值的范围是 - 7f ~ 7f（表示十六进制的大小写字母不加区别），如 0X13，0X6A，- 0X3c。

6 . 某商场在“五一”期间，顾客购物时收费有 4 种情况：普通顾客一次购物累计少于 100 元，按 A 类标准收费（不打折），一次购物累计多于或等于 100 元，按 B 类标准收费（打 9 折）；会员顾客一次购物累计少于 1000 元，按 C 类标准收费（打 8 折），一次购物累计等于或多于 1000 元，按 D 类标准收费（打 7 折）。测试对象是按以上要求计算顾客收费的模块，按照路径覆盖法设计测试用例。

一、填空题

- 1 . 程序设计语言的特性主要有心理特性、工程特性和 技术特性 三个方面。
- 2 . 程序语言的工程特性主要表现为可移植性、 开发工具的可利用性 、软件的可重用性和可维护性。
- 3 . 为了提高程序的易读性，同时为减少错误，提高软件开发效率，编码时应注意养成良好的 程序设计 风格 。
- 4 . 程序加注释对理解程序提供了明确指导，根据作用不同注释分 序言性注释 和 功能性注

释_____。

5. 软件需求分析之后，软件的设计、编码、测试与语言的特性有很大的关系，这个特性主要是语言的_____技术_____特性。

6. 语句构造的原则是_____简单直接_____，不能为了追求效率而使代码复杂化。

7. FORTRAN语言是世界上第一个被正式推广应用的计算机语言，它不仅面向科学计算，数据处理能力也极强。

8. 汇编语言是面向机器的，可以完成高级语言无法满足要求的特殊功能，如与外部设备之间的一些接口操作。

9. 为了使数据定义更容易理解和维护，一个语句说明多个变量时，各变量名按_____字典_____排列。

10. 编码是将详细设计得到的处理过程的描述转换为基于某种计算机语言的程序。

二、选择题

1. 结构化程序设计主要强调的是（ B ）

A.程序的执行效率 B 程序的易读性 C 程序的规模 D 程序可移植性

2. 下列选项中，不属于程序设计语言心理特性的表现形式的是（ C ）。

A 简洁性 B 歧义性 C 可移植性 D 局部性和顺序性

3. 下列选项中与选择程序设计语言无关的因素是（ A ）。

A 程序设计风格 B 软件执行的环境 C 软件开发的方法 D 项目的应用领域

4. 项目应用领域是选择程序设计语言的关键因素，下列不属于项目应用领域类型的选项是（ D ）。

A 系统软件 B 数据处理和数据库应用 C 人工智能 D.UNIX 操作系统

5. 不适合做科学工程计算的语言是（ D ）。

A.FORTRAN 语言 B.C 语言 C.Pascal 语言 D.Lisp 语言

6. 源程序中的序言性注释应置于每个模块的其始部分，该注释内容不应有（ B ）。

A 模块的用途 B 语句的功能 C 数据描述 D 模块的功能

7. 下列适用于开发人工智能领域的语言是（ C ）

A Ada 语言 B C 语言 C Prolog 语言 D Cobol 语言

8. 程序设计语言的工程特性其中之一表现在（ B ）

A 开发工具的可利用性 B 简洁性

C 数据结构的描述性 D 数据库的可操作性

9. 下列不合适作为数据处理与数据应用的语言是（ A ）

A Cobol 语言 B SQL 语言 C 4GL D Ada 语言

10. 提高程序效率的根本途径并非在于 (D)

A 选择良好的设计方法 B 选择良好的数据结构

C 选择良好的算法 D 对程序语句做调整

三、简答题

1、程序语言有哪些共同特征 ？

答：程序语言的共同特征 (似为特性更符合题意)是指程序设计语言的语言特性，包括心理特性、工程特性和技术特性三个方面。语言的心理特性对人机通信的质量有主要的影响；语言的工程特性对软件开发成功一否有重要的影响，此外，语言的技术特性也会影响软件设计的质量。

2、在项目开发时选择程序设计语言通常考虑哪些因素 ？

答：通常要考虑的因素有：

项目的应用领域、软件开发的方法、软件执行的环境、算法和数据结构的复杂性以及软件开发人员的知识等。

3、第 4 代语言 (4GL)有哪些主要特征 ？

答：4GL 的主要特征是：

(1)友好的用户界面

(2)兼有过程性和非过程性双重特性

(3)高效的程序代码

(4)完备的数据库

(5)应用程序生成器

4、举例说明各种程序设计语言的特点及适用范围 ？

答：以常见的一些编程语言为例：

FORTRAN 语言，它不仅面向科学计算，数据处理能力也极强。适用于科学工程计算

Pascal 语言，具有很强的数据和过程结构化能力，它语言简明，数据类型丰富，程序结构严谨。它是第一个体现结构化编程思想的语言。用于描述结构化算法和科学计算 ,操作系统的编写

C 语言：功能强大，支持复杂的数据结构，可大量运用指针，具有丰富灵活的操作运算符及数据处理操作符，它还具有汇编语言的某些特征，使程序运行效率高。可用于许多领域，特别是在操作系统等方面有较

大优势。

Coble 语言，它具有极强的数据定义能力，程序说明与硬件环境说明分开、数据描述与算法描述分开。结构严谨层次分明，广泛用于商业数据处理。

SQL：结构化查询语言，用于对数据库进行存取管理。

汇编语言和 Ada 语言，主要用于实时处理和编写系统软件。

Prolog、Lisp 语言。Lisp 是一种函数型语言，特别适用于组合问题中符号运算和表处理，用于定理证明、树的搜索和其他问题的求解。Prolog 是一种逻辑型语言，它提供了支持知识表示的特性，每一个程序由一组表示事实、规则和推理的句子组成，比较接近与自然语言。这两种语言主要用于人工智能系统开发。

C++、Java 语言，是面向对象的设计语言，用于面向对象的软件开发。

5、什么是程序设计风格？为了具有良好的设计风格，应注意哪些方面的问题？

答：程序设计风格是指一个人编制程序时所表现出来的特点、习惯、思路等。

要形成良好的程序设计风格，应从源程序文档化、数据说明、语句构造、输入输出和追求效率几个方面加以注意。

一、填空题

1．软件原来没有帮助信息，使用不方便，现在要增加帮助信息。这种维护性活动称之为 完善性维护。

2．调试也称 纠错，是在成功的测试之后才开始进行，其目的是确定错误的原因 和位置，并改正错误。

3．调试技术包括 简单调试、归纳法调试、演绎法调试和 回溯法调试。

4．回溯法调试是从 程序产生错误的地方出发，而归纳法调试是从 测试结果发现的线索入手。

二、选择题

1.某个应用软件原来是在 DOS 环境下运行的，现在要把它移植到 Windows 环境下来运行，这种修改软件的过程称为（ B ）

A 校正性维护 B 适应性维护 C 完善性维护 D 预防性维护

2.产生软件维护的副作用是指（ D ）

A 开发软件时的错误 B 运行时的错误

C 隐含的错误 D 因修改软件而造成的错误

3. 软件维护是指（ C ）

A 维护软件的正常运行

B 软件的配置更新

C 对软件的改进，适应和完善

D 软件开发期的一个阶段

三、简答题

1、软件维护有哪些内容？

a、校正性维护 b、适应性维护 c、完善性维护 d、预防性维护

2.软件维护的特点是什么？

a、非结构化维护和结构化维护

b、维护的困难性

c、软件强维护的费用

3.软件维护的流程是什么？

a、制定申请维护报告 b、审查申请报告并批准 c、进行维护并作详细记录 d、覆审

4.软件维护的副作用有哪些？

编码、数据、文档副作用

5.什么是软件的可维护性？可维护性的度量的特性是什么？

软件的可维护性：软件能够被理解、校正、适应及增强功能的容易程度。

可维护性的度量的特性是：可理解性、可测试性、可修改性、可靠性、可移植性、可使用性和效率。

6.提高可维护性的方法有哪些？

A、建立明确的软件质量目标。 B、利用先进的软件开发技术和工具。

C、建立明确的质量保证工作。 D、选择可维护的程序设计语言。

E、改进程序文档。