



# 1. 什么是软件工程？请叙述软件生命周期的各个阶段。

**软件工程**是应用工程化方法来开发、运行和维护软件的学科，目的是通过规范化的过程、技术和工具来提高软件的质量、可维护性和开发效率。

**软件生命周期**包括以下阶段：

1. **需求分析：**
    - 确定用户需求和系统功能。
    - 结果：需求文档。
  2. **系统设计：**
    - 设计软件的整体结构，包括模块划分、接口设计等。
    - 结果：设计文档。
  3. **编码实现：**
    - 将设计转化为可执行的代码。
    - 结果：源代码。
  4. **测试：**
    - 验证软件是否满足需求，常包括单元测试、集成测试和系统测试。
    - 结果：测试报告。
  5. **部署：**
    - 将软件交付用户并安装运行。
  6. **运行与维护：**
    - 修复缺陷、优化性能、更新功能。
- 

# 2. 为什么要提出软件开发模型的概念？请叙述瀑布模型所包含的6个工程活动。

**提出软件开发模型的原因：**

- 规范开发流程，使软件开发更加有序。
- 提高开发效率，减少重复劳动。
- 便于沟通协作，明确各阶段任务。
- 提供风险控制机制，避免开发中断或失败。

**瀑布模型的6个工程活动：**

1. **需求分析**：明确用户需求，形成需求规格说明书。
  2. **系统设计**：设计系统架构、模块分解等。
  3. **实现（编码）**：根据设计文档开发源代码。
  4. **测试**：验证功能是否正确，确保无重大缺陷。
  5. **部署与交付**：将软件安装并交付用户使用。
  6. **维护**：对软件进行修复、优化和扩展。
- 

### 3. 需求分析的目的是什么，有什么作用？

**目的：**

- 明确用户实际需求。
- 确定软件应实现的功能和约束。

**作用：**

1. **提供开发依据**：为设计和开发提供明确的方向。
  2. **降低后期成本**：减少因需求不清导致的返工。
  3. **确保用户满意**：保证开发的软件符合用户预期。
  4. **便于协作**：为开发团队和用户之间建立共识。
- 

### 4. 模块独立性的两个度量标准是什么？

**模块独立性**通过以下两个标准进行度量：

1. **内聚性（Cohesion）**：
    - 描述模块内部元素之间的联系强度。
    - 内聚性越高，模块的独立性越强。
  2. **耦合性（Coupling）**：
    - 描述模块与其他模块之间的依赖程度。
    - 耦合性越低，模块的独立性越强。
-

## 5. 内聚和耦合的含义是什么？

### 1. 内聚 (Cohesion) :

- 描述模块内部功能的相关性。
- 高内聚意味着模块职责单一，结构清晰。
- 示例：一个模块专门处理用户登录，而不涉及其他功能。

### 2. 耦合 (Coupling) :

- 描述模块之间的依赖关系。
  - 低耦合意味着模块之间的联系松散，易于独立修改。
  - 示例：模块A调用模块B的接口，而不是直接访问模块B的内部数据。
- 

## 6. 什么是白盒测试？什么是黑盒测试？

### 1. 白盒测试 (White-box Testing) :

- **定义**：测试人员基于软件的内部逻辑和代码结构设计测试用例。
- **特点**：需要了解代码，测试覆盖所有逻辑路径。
- **适用场景**：单元测试、代码逻辑复杂的模块。
- **示例**：验证所有条件语句的分支是否都被测试。

### 2. 黑盒测试 (Black-box Testing) :

- **定义**：测试人员不需要了解软件内部逻辑，仅依据需求说明书设计测试用例。
- **特点**：关注输入和输出的正确性。
- **适用场景**：功能测试、用户界面测试。
- **示例**：输入一组用户名和密码，检查登录功能是否正确。