

软件工程概述

计算机软件

软件定义

- IEEE定义：软件是计算机程序、规程以及运行计算机系统所需要的文档和数据
- 另一种公认的解释：软件是包括程序、数据及其相关文档的完整集合

软件特点

- 软件是一种具有抽象的逻辑实体：无法直接看到形态，必须通过观察、分析、思考或实际运行使用，才能了解软件功能、性能
- 软件的开发是一种逻辑思维成熟的过程，而无明显制造过程
- 软件没有磨损和老化问题，但存在软件退化问题：用户需求不断变化，反复修改的软件稳定性和可维护性越来越差，可用性越来越低
- 软件的开发依然很原始：未完全摆脱手工艺的开发方法
- 软件是高度复杂的逻辑题：实际需求的复杂性和程序逻辑的复杂性
- 软件研制成本高，成本比例逐年增加
- 软件投入运行时还涉及到许多社会因素

软件分类

- 根据软件完成功能所处层次
 - 系统软件
 - 应用软件
 - 中间件软件
- 根据软件服务对象范围
 - 通用软件：操作系统、数据库等
 - 定制软件：企业ERP等
 - 可配置软件

软件危机

软件危机表现

- 软件开发计划难以制订：缺乏经验，成本和进度估计不准确
- 软件开发费用和进度失控
- 软件产品无法让用户满意
- 软件产品质量难以保证
- 软件通常是不可维护的：错误难以修正，修正错误带来新错误
- 软件通常没有适当的文档资料
- 软件成本在计算机系统总成本中所占比例逐年上升

软件工程

软件工程定义

应用系统化、规范化、定量的方法来开发、运行和维护软件，即将工程应用到软件

软件工程要素

- 方法：提供“如何做”的技术
- 工具：提供自动或半自动的软件支撑环境
- 过程：将软件工程的方法和工具综合起来已达到合理、及时的进行计算机软件开发的目的

软件工程目标

在规定时间、费用的条件下生产具有正确性、可用性以及开销适宜的软件产品。软件应具有以下性质：

- 可修改性
- 有效性：有效的利用计算机资源
- 可靠性：具有挽回因操作不当造成软件失效的能力
- 可理解性
- 可维护性：能够修改和改进
- 可重用性
- 可适应性：采用广为流行的语言，在广为流行的操作系统上运行，采用标准的术语和格式书写文档
- 可移植性
- 可追踪性：记录指令现场，以便追踪问题的因果关系
- 可互操作性：多个软件元素相互通信并协同完成任务的能力

最终目的：摆脱手工生产软件的状况，逐步实现软件研制和维护的自动化

软件工程研究的内容

- 软件开发技术
 - 软件开发方法学
 - 开发过程模型
 - 开发工具
 - 软件工程环境
- 软件工程管理
 - 软件管理学
 - 软件工程经济学
 - 软件心理学

软件工程的原则

- 选取适宜的开发模型
- 采用合适的设计方法
- 提供高质量的工程支持力度
- 重视开发过程的管理

软件工程原理

- 一般原理
 - 抽象
 - 信息隐藏

- 模块化
- 局部化
- 确定性
- 一致性
- 完备性
- 可验证性
- 基本原理
 - 用分阶段的生命周期计划严格管理
 - 坚持进行阶段评审
 - 实行严格的产品控制
 - 采用现代化程序设计技术
 - 结果应能清除的审查
 - 开发小组的人员应少而精
 - 承认不断改进软件工程实践的意义

软件工程知识体系知识域

- 软件需求
- 软件设计
- 软件构造
- 软件测试
- 软件维护
- 软件配置管理
- 软件工程管理
- 软件工程过程
- 软件工程工具与方法
- 软件质量