

BUPT
TSEG

软件工程 模型与方法

Models & Methods of SE

软件实现

田野: yetian@bupt.edu.cn

本章内容

- 软件实现的目标与任务
- 程序语言的选择
- 源程序文档化

目标与任务

- 软件实现是软件详细设计的后续阶段及任务，即程序编码；
- 程序编码需要根据具体情况条件确定具体的程序设计语言；
- 按照详细设计及概要设计的要求转换成选定的编程语言；
- 进一步按照要求进行必要的软件单元测试，使可执行程序达到软件的质量要求。

程序语言的选择

- 参考应用领域;
- 根据用户的要求;
- 参考现有的工具及环境;
- 程序员的能力水平;
- 可移植性的要求;

源程序文档化

- 从软件工程的角度，对于源程序除了质量要求之外，为了后期代码的维护和更改，还必须从提高可阅读性，即达到源程序文档化。
 - 标识符命名：模块名（类名及方法名），变量/常量名
 - 名称需清楚表示具体的含义，采用添加前缀和后缀增加可阅读性；
 - 使用专业术语（业务词汇），注意编码语言的关键字冲突；
 - 注意大小写及长度；
 - 源程序布局
 - 编码之前定义统一的编码规范；
 - 规定合理的注释、缩进、空格、空行等方式；
 - 程序注释
 - 序言性注释
 - 功能性注释

序言性注释

- 注释的目的：解释程序的主要内容及难点说明；
- 序言性注释：位于程序代码之前，说明该模块（类及方法）具体作用
- 主要包括以下内容：
 - 程序标题：模块名称
 - 模块描述：该模块的功能和目的说明
 - 主要算法：（option）说明算法结构
 - 接口说明：说明该模块与其他模块的调用关系
 - 开发简历：
 - 创建者、创建时间；
 - 修改者、修改时间、修改内容；
 - 版本

功能性注释

- 在序言性注释的基础上，对于程序体中复杂难于理解的程序结构进行局部说明；
 - 主要描述一段程序，必要时对某一段进行说明；
 - 修改代码的同时，对应的功能性注释也要进行修改；