

第5章内容概要

- 软件设计过程
- 软件设计原理
- 启发规则

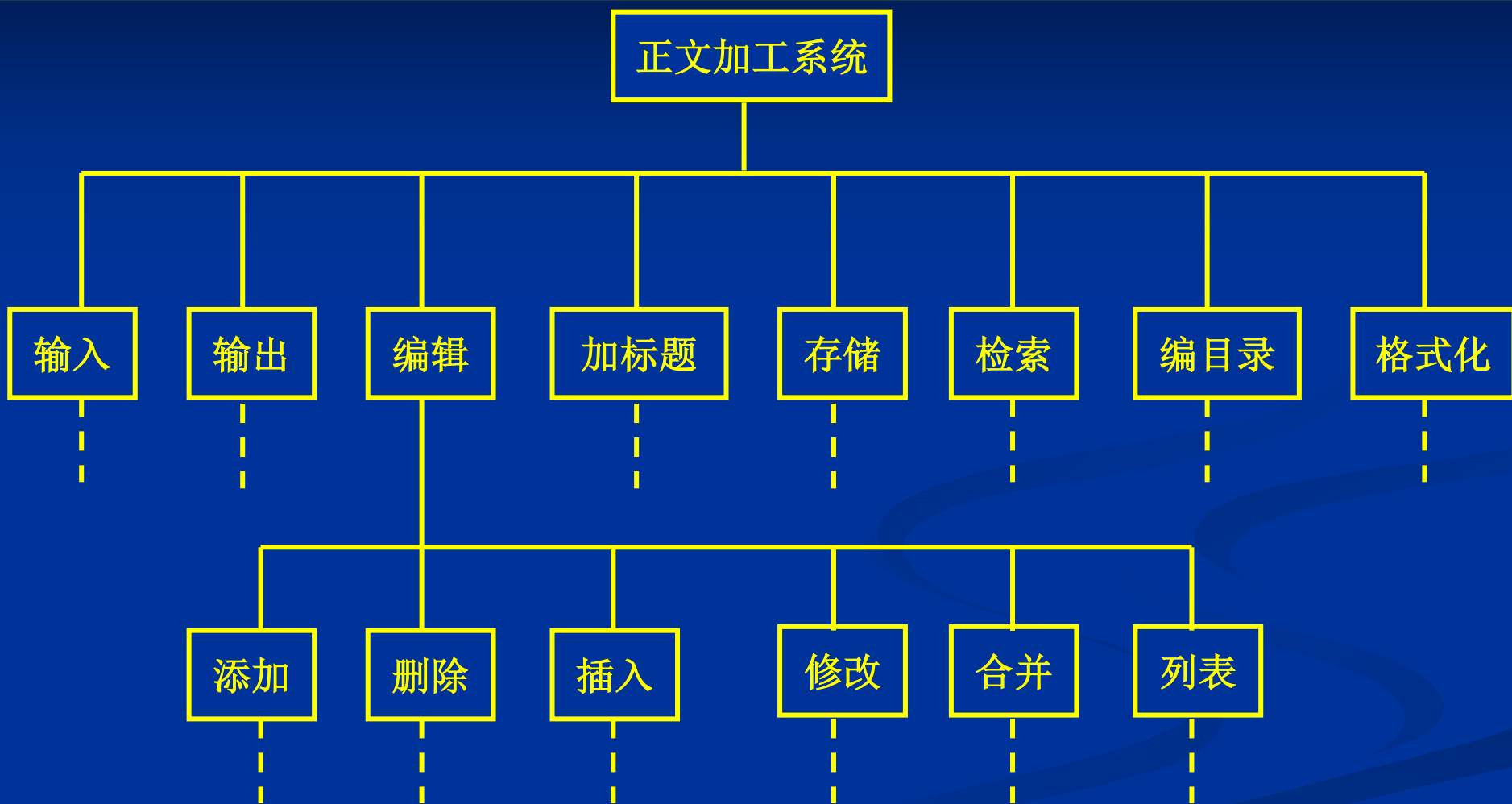
★ ■ 描绘软件结构的图形工具

- 面向数据流的设计方法
- 软件体系结构

描绘软件结构的图形工具

- **层次图**：用来描绘软件的层次结构。
- 层次图中的一个矩形框代表一个模块，方框间的连线表示调用关系。
- 层次图很适于在自顶向下设计软件的过程中使用。

描绘软件结构的图形工具

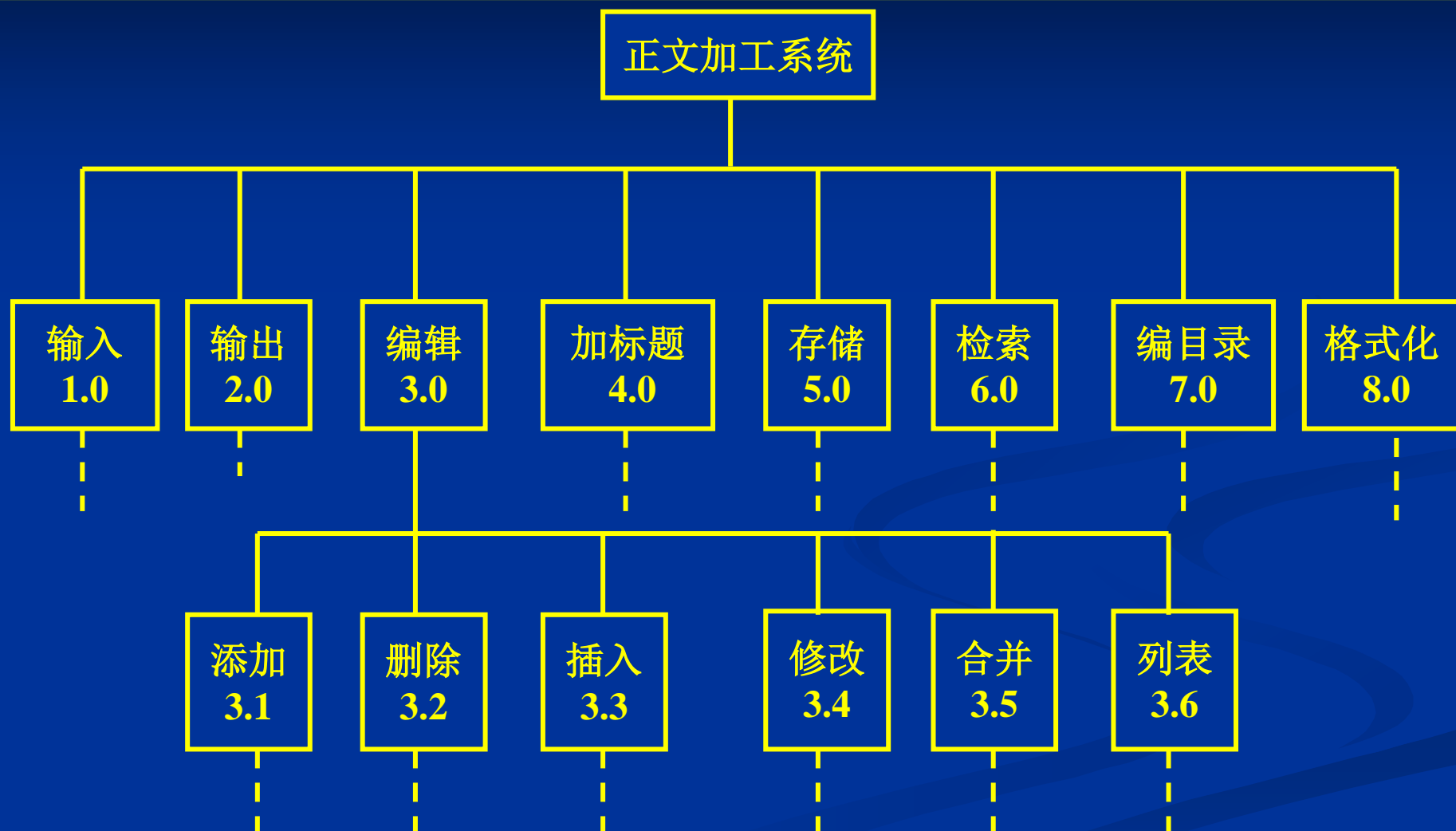


一个正文加工系统的层次图

描绘软件结构的图形工具

- **HIPO图**：是美国IBM公司发明的“层次图加输入 / 处理 / 输出图”的英文缩写。
- 为增加可追踪性，在层次图部分除最顶层方框外，每个方框都要加上编号。
- 和层次图中每个方框相对应，应该有一张IPO图描绘这个方框代表的模块的处理过程。
- HIPO图中的每张IPO图内都应该明显地标出它所描绘的模块在层次图中的编号，以便追踪了解这个模块在软件结构中的位置。

描绘软件结构的图形工具



带编号的层次图（H图）

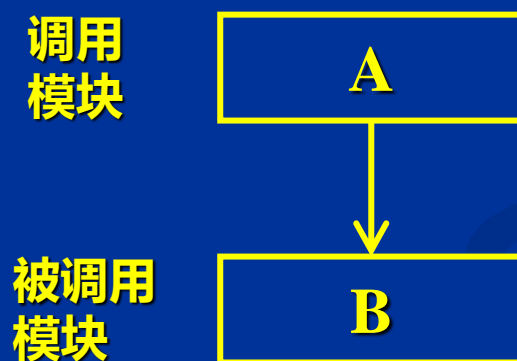
回顾第三章讲到的改进后的IPO图

■ 改进的IPO图

IPO 表	
系统：_____	作者：_____
模块：_____	日期：_____
编号：_____	
被调用：	调用：
输入：	输出：
处理：	
局部数据元素：	注释：

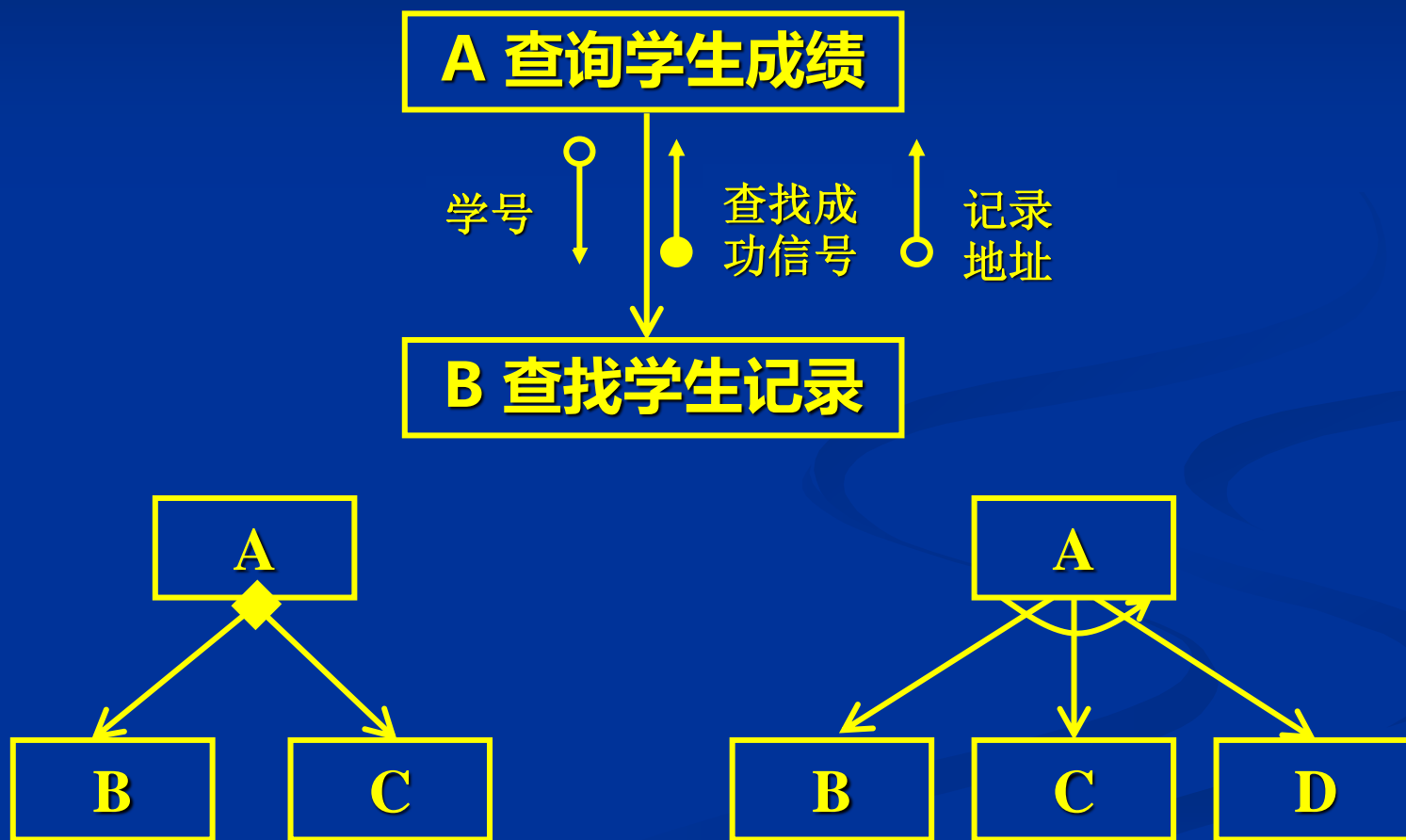
描绘软件结构的图形工具

- 结构图(Structure Chart ,SC): 
 - 主要内容也是模块和模块间的调用关系。



描绘软件结构的图形工具

- 模块间的信息传递与两个辅助符号：



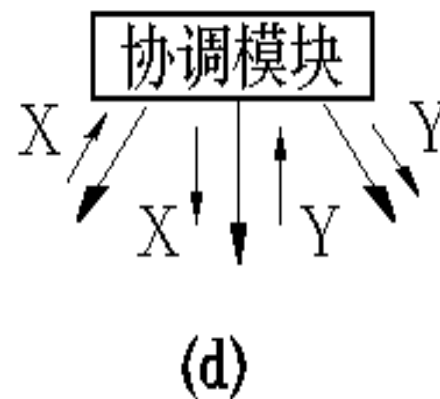
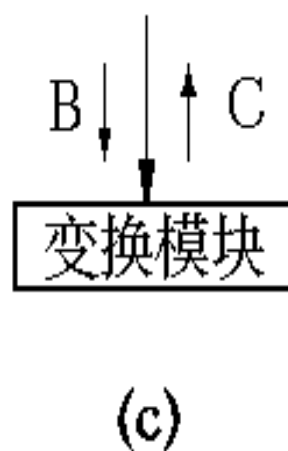
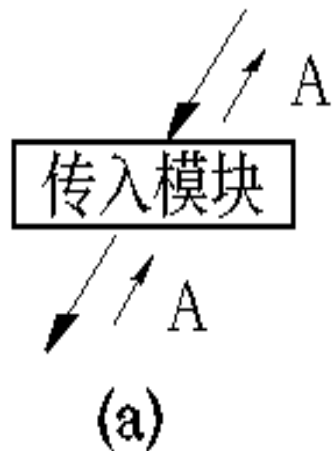
描绘软件结构的图形工具

- 在系统结构图中，不能再分解的底层模块称为原子模块。
- 如果一个软件系统的全部实际加工都由原子模块来完成，而其他所有非原子模块仅仅执行控制或协调功能，这样的系统就是完全因子分解的系统。
- 完全因子分解的系统是理想化的，实际设计中都是尽量向这个目标靠拢。
- 一般地，系统结构图中有4种类型的模块：

描绘软件结构的图形工具

- **传入模块** — 从下属模块取得数据，经过某些处理，再将其传送给上级模块。它传送的数据流叫做逻辑输入数据流。
- **传出模块** — 从上级模块获得数据，进行某些处理，再将其传送给下属模块。它传送的数据流叫做逻辑输出数据流。
- **变换模块** — 它从上级模块取得数据，进行特定的处理，转换成其它形式，再传送回上级模块。它加工的数据流叫做变换数据流。
- **协调模块** — 对所有下属模块进行协调和管理的模块。

描绘软件结构的图形工具



系统结构图的4种模块