

NEW PLAN

软考高级架构师

• — 段 新 征 程 •





大纲介绍





系统设计-业务流程设计

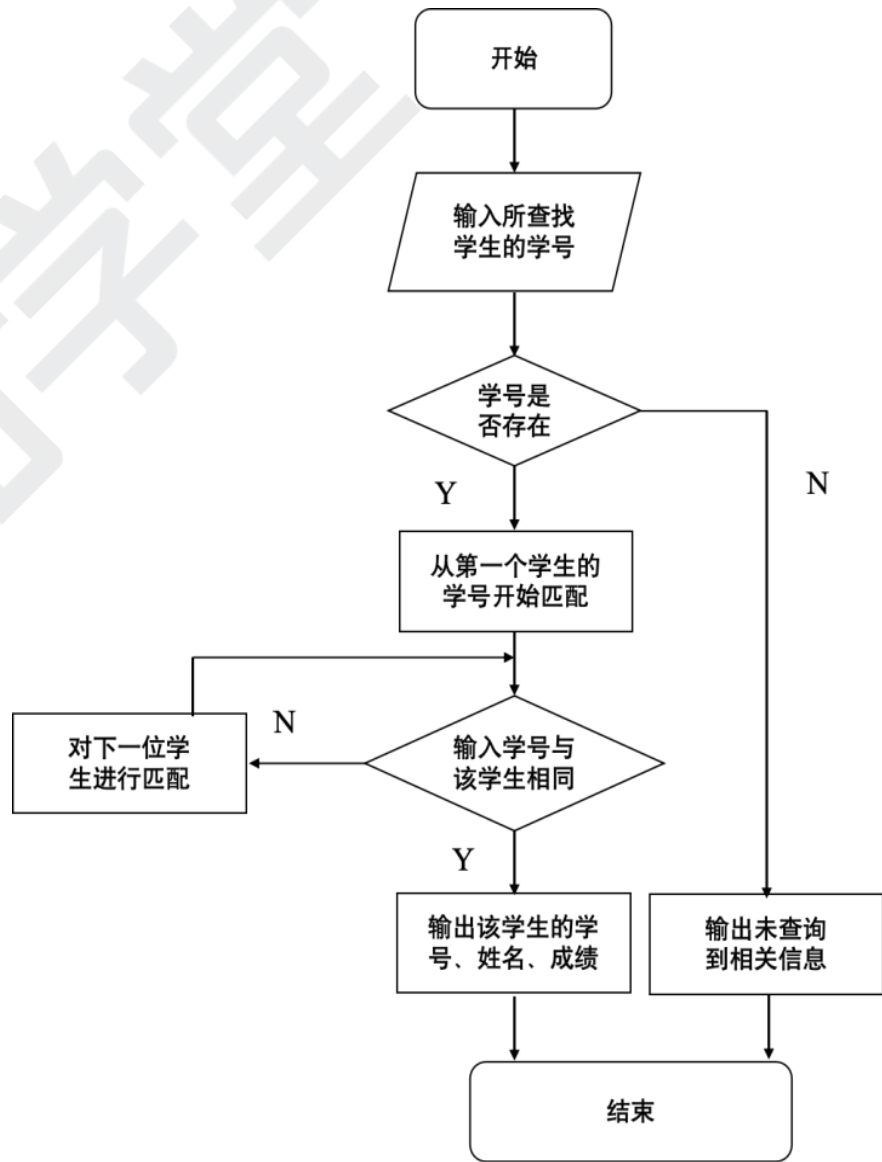
流程设计表示工具:



- 程序流程图(Program Flow Diagram,PFD): 用一些图框表示各种操作, 它独立于任何一种程序设计语言, 比较直观、清晰, 易于学习掌握。任何复杂的程序流程图都应该由顺序、选择和循环结构组合或嵌套而成。



- IPO图: 也是流程描述工具, 用来描述构成软件系统的每个模块的输入、输出和数据加工。前面学习的数据流图其实就是一种IPO图





系统设计-业务流程设计



● N-S图：比较容易表示嵌套和层次关系，并具有强烈的结构化特征。但是当问题很复杂时，**N-S图可能很大，因此不适合于复杂程序的设计**



● 问题分析图(PAD)：是一种**支持结构化程序设计的图形工具**。PAD具有清晰的逻辑结构、标准化的图形等优点，更重要的是，它引导设计人员使用结构化程序设计方法，从而提高程序的质量。

例如：将数组 A(1)到 A(10)从大到小进行选择法排序的算法描述如图 5.4 所示。

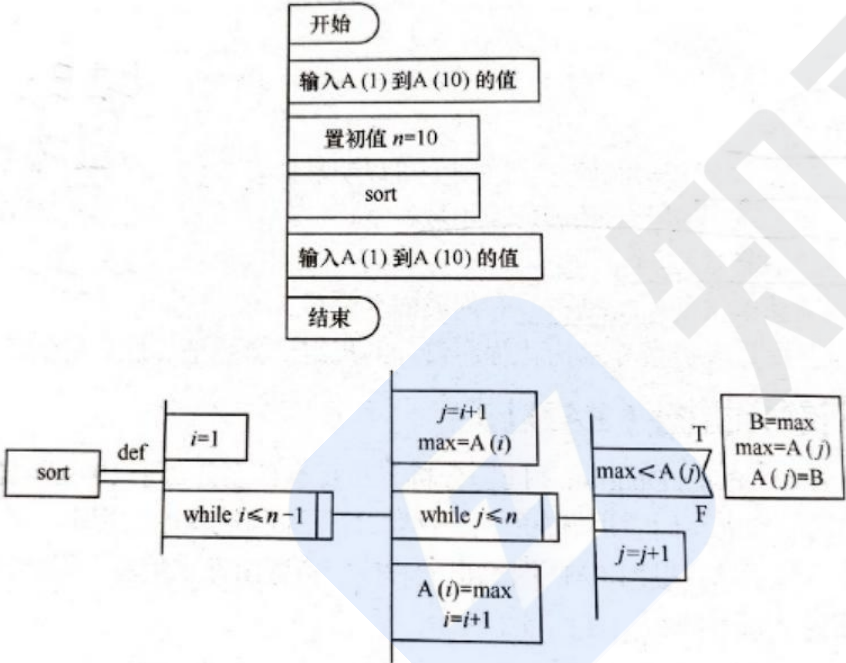
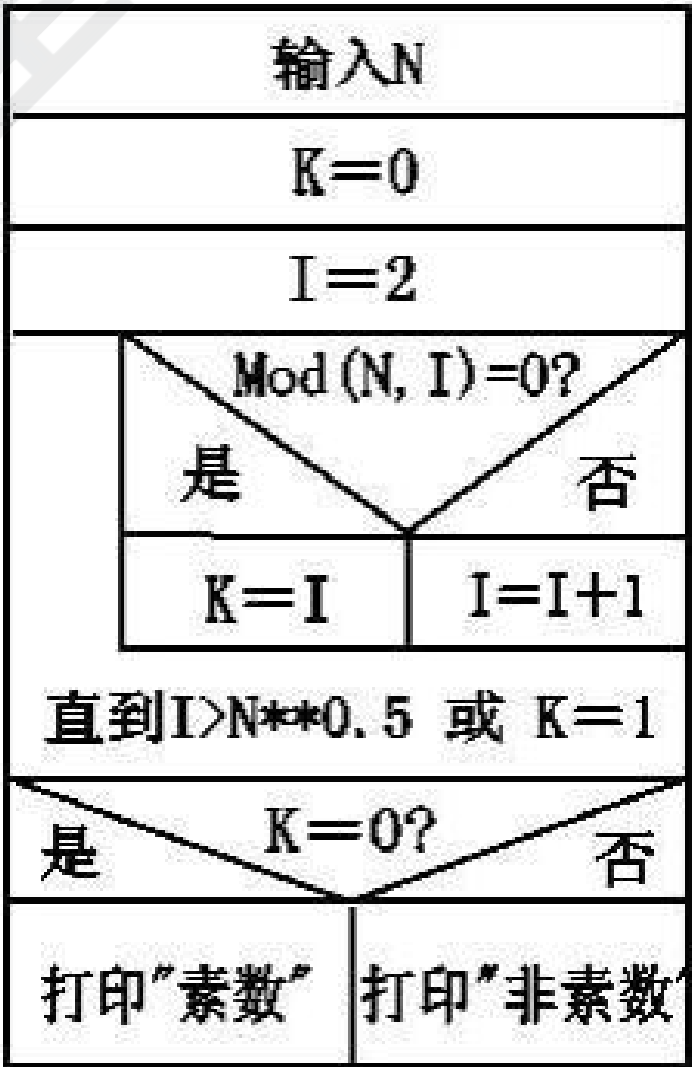


图 5.4 PAD 图示例





系统设计-业务流程设计分类



业务流程管理(BPM)

- 是一种方法论，用于优化、管理和自动化组织内的业务流程。它涉及识别、建模、分析和优化业务流程，以实现更高效、灵活和协调的运作。BPM关注整个业务流程的管理，包括流程的执行、监控、优化和自动化



业务流程重组BPR

- BPR是更为激进的方法，它着眼于对现有业务流程进行彻底的重新设计和改造，以实现质的飞跃的改进，是对企业的业务流程进行根本性的再思考和彻底性的再设计，从而获得可以用诸如成本、质量、服务和速度等方面的业绩来衡量的显著性的成就。

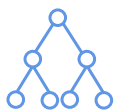


流程管理包含三个层面：规范流程、优化流程和再造流程



流程设计的任务是设计出系统所有模块和它们之间的相互关系，并具体设计出每个模块内部的功能和处理过程。以下关于流程设计的叙述，正确的是()。

- A.任何复杂的程序流程图都应该由顺序、选择、循环结构构成
- B.IPO图不适合用来进行流程设计
- C.PAD图是一种支持原型化设计方法的图形工具
- D.N-S图容易表示嵌套关系和层次关系，特别适合于设计非常复杂的流程答案



业务流程重组(Business Process Reengineering,BPR)是针对企业业务流程的基本问题进行回顾，其核心思路是对业务流程的()改造，BPR过程通常以()为中心。

- | | | | |
|-------|-------|-------|-------|
| A.增量式 | B、根本性 | C、迭代式 | D、保守式 |
| A.流程 | B、需求 | C、组织 | D、资源 |



系统设计



系统设计主要目的：为系统制定蓝图，在各种技术和实施方法中权衡利弊，精心设计，合理地使用各种资源，**最终勾画出新系统的详细设计方法**。简单点说，系统设计就是告诉你该怎么做，需求分析就是找到你要做什么，系统设计就是告诉你怎么做



系统设计方法：**结构化设计方法，面向对象设计方法。**



系统设计的主要内容：**概要设计、详细设计。**



概要设计基本任务：又称为**系统总体结构设计**，是将系统的功能需求分配给软件模块，确定每个模块的功能和调用关系，**形成软件的模块结构图，即系统结构图**。还需要产出概要设计说明书



详细设计的基本任务：**模块内详细算法设计、模块内数据结构设计、数据库的物理设计、其他设计(代码、输入/输出格式、用户界面)、编写详细设计说明书、评审。**



系统设计



系统设计基本原理： 当进行系统设计时，有一些基本原则可以帮助我们更好地规划和构建软件系统。以下是这些原则的通俗解释：

- **抽象化：** 就像画家将一幅画分成不同的图层，系统设计中也会将复杂的问题分成更小、更易管理的部分。这样做能让我们关注重要的细节，同时不被不必要的复杂性所困扰。
- **自顶而下，逐步求精：** 想象你在建立一座高楼大厦，你首先会构想整体的结构，然后一层一层地逐步建造。在系统设计中也是如此，我们先考虑整体框架，然后逐步添加细节，以确保每个部分都被仔细设计和优化。
- **信息隐蔽：** 想象你开车时并不需要了解引擎的每个零件如何运作。在系统设计中，我们可以隐藏复杂的内部细节，只暴露必要的信息给其他部分。这样能够降低不同模块之间的相互影响，使系统更易于管理和维护。
- **模块独立（高内聚，低耦合）：** 就像用乐高积木一样，系统设计中的各个部分应该是独立的，即每个模块都有自己的功能并且尽可能不依赖其他模块。高内聚表示模块内部的元素彼此紧密相关，低耦合表示模块之间的依赖尽可能降低。



系统设计



系统设计基本原则：当进行系统设计时，一些原则可以帮助我们构建清晰、可维护且高效的软件系统。以下是这些原则的解释：

- 保持模块的大小适中：就像把一块大蛋糕切成适合一口吃的小块一样，系统中的模块也应该适中，不要太大也不要太小。这样做可以使每个模块的功能清晰，并且易于理解、测试和维护。
- 尽可能减少调用的深度：想象你需要找到某本书，如果每次你要找的书都在另一本书的页码里，你需要不断翻书才能找到。在系统设计中，减少调用深度意味着尽量减少程序的层层嵌套，以避免复杂的调用关系，使代码更清晰和高效。
- 多扇入，少扇出：想像一个火车站，火车到站后乘客下车，站台上的人上车，但火车的出口只有一个。在系统设计中，模块应该允许多个模块调用它（多扇入），但它本身应尽量少调用其他模块（少扇出），以减少模块之间的复杂关系。
- 单入口，单出口：就像你进入一座建筑物只有一个门一样，在模块设计中，最好只有一个入口和一个出口。这样可以使模块的功能和接口更清晰，不容易引入混淆和错误。
- 模块的作用域应该在模块之内：就像一个厨房用具只在厨房里使用一样，模块的功能应该限制在模块内部，不要让外部模块直接访问内部的细节。这有助于封装和隐藏内部逻辑，防止外部干扰。
- 功能应该是可预测的：就像按下开关会打开灯一样，模块的功能应该是可预测的。使用者在调用一个模块时应该能够合理地预期它会发生什么，从而避免产生混淆和不可预知的结果。

 系统设计基本原理：抽象、模块化、信息隐蔽、模块独立（高内聚和低耦合）。

 衡量模块独立程度的标准有两个：耦合性和内聚性。内聚程度从低到高如下表所示：

内聚分类	定义	记忆关键字
偶然内聚	一个模块内的各处理元素之间没有任何联系。	无直接关系。
逻辑内聚	模块内执行若干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。	逻辑相似、参数决定。
时间内聚	把需要同时执行的动作组合在一起形成的模块。	同时执行。
过程内聚	一个模块完成多个任务，这些任务必须按指定的过程执行。	指定的过程顺序。
通信内聚	模块内的所有处理元素都在同一个数据结构上操作，或者各处理使用相同的输入数据或者产生相同的输出数据。	相同数据结构、相同输入输出。
顺序内聚	一个模块中的各个处理元素都密切相关于同一功能且必须顺序执行，前一个功能元素的输出就是下一个功能元素的输入。	顺序执行、输入为输出。
功能内聚	最强的内聚，模块内的所有元素共同作用完成一个功能，缺一不可。	共同作用、缺一不可。



耦合程度从低到高如下表所示

耦合分类	定义	记忆关键字
无直接耦合	两个模块之间没有直接的关系，它们分别从属于不同模块的控制与调用，不传递任何信息。	无直接关系。
数据耦合	两个模块之间有调用关系，传递的是简单的数据值，相当于高级语言中的值传递。	传递数据值调用。
标记耦合	两个模块之间传递的是数据结构。	传递数据结构。
控制耦合	一个模块调用另一个模块时，传递的是控制变量，被调用模块通过该控制变量的值有选择的执行模块内的某一功能。	控制变量、选择执行某一功能。
外部耦合	模块间通过软件之外的环境联合（如 I/O 将模块耦合到特定的设备、格式、通信协议上）时。	软件外部环境。
公共耦合	通过一个公共数据环境相互作用的那些模块间的耦合。	公共数据结构。
内容耦合	当一个模块直接使用另一个模块的内部数据，或通过非正常入口转入另一个模块内部时。	模块内部关联。



人机界面设计



人机界面设计三大黄金原则:

置于用户控制之下

- 以不强迫用户进入不必要的或不希望的动作的方式来定义交互方式
- 提供灵活的交互
- 允许用户交互可以被中断和撤消
- 当技能级别增加时可以使交互流水化并允许定制交互
- 使用户隔离内部技术细节
- 设计应允许用户和出现在屏幕上的对象直接交互

减少用户的记忆负担

- 减少对短期记忆的要求
- 建立有意义的缺省
- 定义直觉性的捷径
- 界面的视觉布局应该基于真实世界的隐喻
- 以不断进展的方式揭示信息

保持界面的一致性

- 允许用户将当前任务放入有意义的语境
- 在应用系列内保持一致性
- 如过去的交互模型已建立起了用户期望，除非有迫不得已的理由，不要改变它



考试真题



某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，前一处理元素的输出就是下一处理元素的输入，则该模块的内聚类型为()内聚

A.过程

B.时间

C.顺序

D.逻辑



已知模块A给模块B传递数据结构X,则这两个模块的耦合类型为()

A.数据耦合

B.公共耦合

C.外部耦合

D.标记耦合



考试真题



系统设计是根据系统分析的结果，完成系统的构建过程。系统设计的主要内容包括();系统总体结构设计的主要任务是将系统的功能需求分配给软件模块，确定每个模块的功能和调用关系，形成软件的()

A.概要设计和详细设计

B.架构设计和对象设计

C.部署设计和用例设计

D.功能设计和模块设计

A.用例图

B.模块结构图

C.系统部署图

D.类图



以下关于软件系统模块结构设计的叙述中，正确的是()。

A.当模块扇出过大时，应把下级模块进一步分解为若干个子模块

B.当模块扇出过小时，应适当增加中间的控制模块

C.模块的扇入大，表示模块的复杂度较高

D.模块的扇入大，表示模块的复用程度高

T H E E N D

功不唐捐，玉汝于成！

• 开 启 新 征 程 •

