

软件工程教学与计算思维能力培养

郭咏梅

(长治学院计算机系, 山西 长治 046011)

摘要: 针对学生思维能力和解决实际问题能力弱的问题, 分析了软件工程课程的基本知识点和计算思维能力点之间的关联, 探讨了软件工程教学过程中, 有意识、有计划地进行计算思维能力培养的策略和方法。

关键词: 软件工程; 计算思维; 能力培养

中图分类号: TP311.5 **文献标识码:** A

Software Engineering Teaching and Training of Computational Thinking Ability

GUO Yongmei

(Department of Computer Science Changzhi University, Changzhi 046011, China)

Abstract: For the student's inability of thinking and solving practical problems, the paper analyzes the relevance between the basic knowledge in the software engineering course and computational thinking ability, and discusses the methods and strategies of training computational thinking ability during the software engineering teaching process.

Keywords: software engineering; computational thinking; ability training

1 引言(Introduction)

在当前软件产业迅猛发展的形势下, 社会和企业对软件人才的要求越来越高, 不仅要求毕业生有扎实的专业基础知识, 还要有解决实际问题的思维能力和创新能力。但是, 大多数毕业生走向工作岗位后存在理论与实践结合能力较弱和综合应用知识能力欠缺问题^[1], 对实际问题的求解没有清晰的思路, 不能在短时间内胜任企业的工作。因此, 我们应重视和加强培养学生解决现实问题的思维能力和综合应用知识的创新能力。

《高等学校计算机科学与技术专业人才培养》中指出, 计算机专业学生的专业基本能力主要包括计算思维能力、算法设计与分析能力、程序设计与实现能力、系统能力^[2]。其中计算思维能力是至关重要的, 良好的思维能力是取得成功求解问题的关键。本文结合软件工程课程的特点, 重点探讨培养学生计算思维能力的问题, 分析软件工程课程知识点与计算思维能力点之间的对应关系, 研究如何在软件工程课程的教学过程中渗透计算思维方法, 通过理论教学和实践教学培养学生的计算思维能力, 进而提高学生解决实际问题的能力。

2 计算思维、计算思维能力(Computational thinking, computational thinking ability)

计算思维是运用计算机科学的基础概念去求解问题、

设计系统和理解人类的行为, 它包括了涵盖计算机科学之广度的一系列思维活动^[3]。计算思维是利用计算机解决实际问题的思维方法, 它的本质是抽象(Abstract)和自动化(Automation)。典型的计算思维包括一系列广泛的计算机科学的思维方法: 递归、抽象、分解、关注点分离、约简、迭代、预置和缓存、保护、冗余、容错、回溯、纠错和恢复等等。

计算思维能力是指人们运用计算思维方法思考的能力。对于计算机专业人才的培养, 计算思维能力主要包括问题及问题求解过程的符号表示、逻辑思维与抽象思维、形式化证明、建立模型、实现类计算和模型计算、利用计算机技术等^[2]。这些能力的培养需要以课程知识为载体, 在教师有意识、有计划的引导下, 在不断的实践过程中实施^[4]。

3 软件工程课程知识点与计算思维能力点的关联 (The relationship between knowledge points of software engineering courses and the ability of computational thinking)

软件工程是研究软件开发和软件管理的一门工程学科, 它主要包括技术和管理两方面的内容。在新建本科院校的计算机类专业, 软件工程课程重点讲授软件技术方面的知识, 包括软件工程基本概念、基本原理和基本方法, 强调应用计算机技术解决实际问题的方法和步骤, 使学生掌握软件开发

各个阶段规范的方法、工具和过程^[5]。软件工程知识体系主要有结构化方法和面向对象方法两大部分，下面重点分析两种软件开发方法涉及的主要知识点与计算思维方法、计算思维能力点的关联。

3.1 结构化软件开发方法

结构化软件开发方法的主要过程是：结构化需求分析(SA)、结构化设计(SD)、结构化编码测试(SP)，其主要特点是按功能分解系统，自顶向下逐步分解，将一个复杂系统分解为简单子系统，每个阶段都要遵循规范的方法和步骤，通过严格的评审后才能开始下个阶段的工作。各个阶段的主要知识点为与计算思维能力点的对应如表1所示。

表1 结构化方法主要知识点与计算思维能力点
Tab.1 The knowledge points and computational thinking ability of structural method

	SA	SD	SP
知识点	数据分析(数据模型)	软件结构设计	命名规范
	功能分析(功能模型)	数据设计	编程风格
	状态转换分析(行为模型)	接口设计	测试技术
	数据字典	算法设计	测试工具
能力点	约简	自顶向下逐步细化	符号表示
	抽象	关注点分离	逻辑思维
	分解	分而治之	抽象思维
	建立模型	逻辑思维	回溯、纠错
	逻辑思维	抽象思维	利用计算机技术
	抽象思维	符号表示	
	符号表示		

在结构化方法开发软件过程中，涉及大量的计算思维方法：在需求分析阶段，对实际问题进行简化、抽取问题的本质，用到约简、抽象方法；在概要设计阶段，将复杂问题分解为若干个简单问题，用到关注点分离方法；数据库设计用到抽象和分解的方法；在调试程序的过程中，从发现错误的地方，沿程序的控制流往回追踪分析程序代码，直到找到错误，用到回溯方法等等。在开发软件的实践过程中，能够有效训练学生的计算思维能力。例如：需求分析阶段的数据流程图、实体关系图、状态转换图，设计阶段的层次结构图、算法流程图，编码阶段的程序代码都是分析问题和求解问题的符号表示，分析和设计的过程就是逻辑思维与抽象思维的过程，需求分析过程就是建立模型的过程等等。由此可见，软件工程的知识点与计算思维的能力点是密切关联的。

3.2 面向对象软件开发方法

面向对象软件开发方法的主要过程是：面向对象需求分析(OOA)、面向对象系统设计(OOD)、面向对象编码测试(OOP)。与结构化方法相比，面向对象方法的特点是按对象

和类分解系统，更符合人类认识问题和解决问题的方法和过程。其主要知识点与计算思维能力点的关联如表2所示：

表2 面向对象方法主要知识点与计算思维能力点
Tab.2 The points of knowledge and computational thinking ability of object oriented method

	OOA	OOD	OOP
知识点	对象分析(对象模型)	体系结构设计	命名规范
	功能分析(用例模型)	数据设计	编程风格
	场景转换分析(动态模型)	类设计	类和系统测试
	抽象	子系统设计	测试工具
能力点	分解	自顶向下逐步细化	迭代
	关注点分离	分而治之	回溯、纠错
	建立模型	迭代	符号表示
	逻辑思维	关注点分离	逻辑思维
	抽象思维	逻辑思维	抽象思维
	符号表示	抽象思维	实现类、模型计算
		符号表示	利用计算机技术

面向对象的需求分析过程是建立对象模型、用例模型和动态模型的过程，对象分析过程是一个典型的抽象过程；功能分析将系统抽象和分解为若干个用例，是抽象和分解、关注点分离方法的典型应用；面向对象的设计和实现是在需求分析的基础上多次反复迭代的演化过程，对各个子系统的设计也用到抽象和分解、关注点分离方法。这些计算思维方法的应用对应着问题求解过程的符号表示、逻辑思维与抽象思维、建立模型、实现类计算和模型计算、利用计算机技术等计算思维能力点。在面向对象方法开发软件过程中，应用到大量的计算思维方法，有效的教学过程能够让学生理解、应用计算思维方法，达到训练学生计算思维能力的目的。

4 培养计算思维能力的策略和方法(The strategies and methods of developing computational thinking ability)

以软件工程课程知识点为载体，将计算思维训练融合在教学的各个环节当中，重点从意识、理解、实践三个层次入手，循序渐进的提高学生应用计算思维方法解决实际问题的能力。

4.1 有意识的教学方法让学生认识到计算思维的重要性

从我们有关计算思维调研结果来看，大部分学生不了解计算思维的概念，在分析问题和解决实际问题的过程中，不会应用计算思维方法，即使用到一些计算思维方法也是无意识的。因此，在课堂教学和实践教学过程中，要充分发挥教师的主导作用，教师要将课程知识与实际问题紧密结合，激发学生学习兴趣，通过实际案例有意识地、系统地讲解计算
(下转第54页)