

新工科背景下的C语言人才培养研究

肖 斌¹, 汪 敏², 王 兵¹

(1.西南石油大学计算机科学学院, 四川 成都 610500;

2.西南石油大学电气信息学院, 四川 成都 610500)

摘 要: 新经济形态对计算机人才培养提出了新的要求, 结合C语言程序在人才培养体系中的重要性, 以及当前的教学现状, 将计算思维融入教学中培养学生的分析问题的能力和抽象建模的能力, 使得学生的程序设计思维清晰化、规范化。同时结合MOOC平台实施线上线下的混合式教学, 保证教师实施有效过程监督。

关键词: 计算思维; C程序设计; MOOC; 程序设计实践

中图分类号: TP311.5 **文献标识码:** A

Research on C Language Talent Cultivation under the Background of Emerging Engineering

XIAO Bin¹, WANG Min², WANG Bin¹

(1.College of Computer Science, Southwest Petroleum University, Chengdu 610500, China;

2.College of Electrical Information, Southwest Petroleum University, Chengdu 610500, China)

Abstract: The new economic forms put forward new requirements for the cultivation of computer talents. Combining the importance of C language programs in the personnel training system with the current teaching status, this paper integrates computational thinking into teaching to cultivate students' analytical problems and abstract modeling, making students' programming thinking clear and standardized. At the same time, it is combined with the MOOC platform to achieve online and offline hybrid teaching, to ensure that teachers implement effective process supervision.

Keywords: computational thinking; C programming; MOOC; programming practice

1 引言(Introduction)

伴随着大数据、物联网、人工智能、云计算的飞速发展, 实现了新技术与新产业的深度融合, 产生了第四次工业革命下的工程新业态, 带来了工程科技人才培养的新要求。而程序设计是实现信息技术与新工科的关键, 对未来技术和产业起到引领作用^[1]。因此程序设计的工程实践能力、分析设计能力、创新能力、终身学习能力成为新工科软件工程人才的核心能力^[2]。

2 C程序设计的地位和现状(The status and current situation of C programming)

2.1 C程序设计语言的地位

C语言程序设计主要包含变量、语句、控制流程、数组、指针、结构体、函数等知识点内容的一门面向过程的程序设计课程。该语言很好地支持结构化程序设计, 也是面向对象语言、数据结构的基础。

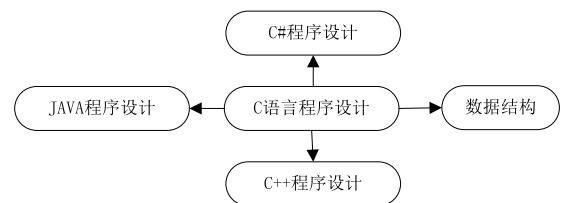


图1 C语言程序设计课程关系图

Fig.1 Relationship among C programming courses

通过C程序设计语言的学习, 学生掌握变量、控制流程语句、函数、数组、指针、结构体的相关基础知识, 掌握程序模块化编程的编程思想, 具备利用计算机的思维方式分析问题并进行数据表达和算法描述的能力、利用基本算法解决问题的能力, 具备工程设计和工程的调试分析能力, 达到利用计算机求解复杂科学问题的计算思维能力。能力培养体系层次关系如图1所示。

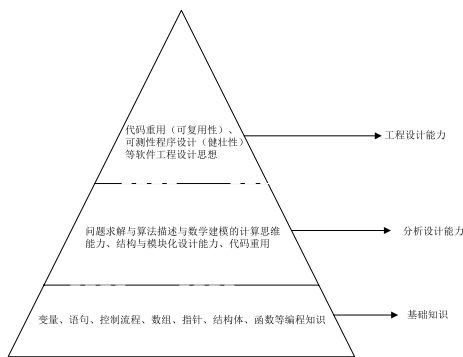


图2 能力培养体系层次图

Fig.2 Hierarchy of the ability training system

2.2 C程序设计语言的教学现状

在新工科背景下，针对C语言程序设计，需要培养基础知识扎实，具有分析设计能力，具有工程设计能力的人才的三层综合能力，但课程实际教学中往往停留在第一层，存在大量问题。

(1)C语言教学中，学生刚从高中升入大学，已有的学习模式与计算机思维的学习模式具有不一致性，记忆性的知识能够学习，但抽象思维和逻辑思维能力比较弱，即计算思维能力较差，导致对问题的分析不透彻，编写程序时不知道如何设置变量；不知道如何设计程序的结构；不知道什么时候定义函数、如何定义函数；不知道函数的粒度和职能范围有多大；不知道数据如何存储、算法如何实现^[3]。学生缺乏这方面的引导和大量训练。

(2)语法规则是记忆性的知识点，但许多教师在教学过程中会花费大量的时间和精力讲述，由于课时有限，当课程结束时，学生只是机械性地记住了这些知识点的规则，但不能真正掌握和理解如何运用这些语法规则，不能利用计算机解决实际问题。高校教育没能完成大学生与社会的接轨。大学的教育要摆脱以传授知识为主，应主要培养计算机专业学生的主动自学能力、抽象思维能力、理解应用能力和实践操作能力。

(3)传统教学导致教学时间有限，无法保证有效的过程监督机制^[4]。课前学习和课后总结测试是保证学习效果的重要途径，在实际教学中，主要依靠学生自觉完成，教师不了解学生的学习状态和学习效果。另外，高校在程序设计类课程中，缺乏资金投入，往往采用大班上课，评阅作业存在有效性和滞后性等问题，而作业是教师掌握学生学习情况的重要途径，导致教师不能全面掌握学生的真实学习效果，不能及时解决学生问题。问题积少成多，学生学习兴趣受到挫败。教师也无法通过有效的作业反馈调整上课内容，改进教学方法。

(4)英文语言基础薄弱遇到计算机语言的初学状态导致学生看不懂错误、理解不了错误，并缺乏分析程序、跟踪程序的能力，不会解决编译时碰到的语法错，更不能解决逻辑错误。这与教学的时间投入、教学的方法，以及学生实践经验缺乏密切相关。学生需要提高阅读程序的能力和调试程序的能力^[5]。

(5)C语言程序设计是一门应用性、实践性和现实性很强的课程，而课程教学内容在创新项目内容深度不够，过多关注程序设计语言本身的训练和数据的简单呈现，而忽视了整体性与综合创新性思维培养。创新实践教学的总体规划、内容设计等方面不合理，实践教学效果差^[6]。

3 C程序设计的教学组织方法(Teaching organization method for C programming)

3.1 计算思维构建C程序设计教学设计

计算思维是指以抽象化与自动化为特征的一种思维。学生之所以不会写程序，不知道怎么写程序就是这种抽象化和自动化能力不足的体现^[7]。通过计算思维能给运用计算机科学的基础概念进行问题求解、系统设计，将增强学生程序设计的能力。其计算思维活动序列图如图3所示。

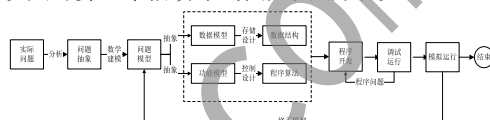


图3 计算思维中问题求解框架图

Fig.3 Framework diagram for problem solving

在图3中，抽象指的是对求解问题的一种描述，是对问题的表达和推演。在C语言中，把实际问题抽象成与其相应的形式化数学模型，然后构建数据结构，确定算法，最后编写程序求解问题。具体而言，抽象就是用变量、函数、数组等数据结构和算法对需要解决的实际问题进行概括描述，抽象的过程实际上就是对实际问题确立数学模型的过程。抽象与实际问题的对应关系，见表1所示。

表1 C语言程序抽象模块

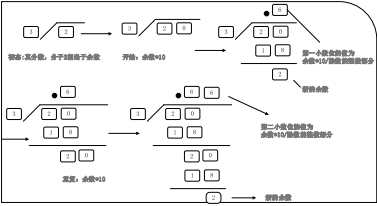
Tab.1 Abstract module

现实世界描述	C语言抽象描述
值不变的特征抽象描述	常量
取值发生变化的特征进行抽象描述	变量
特征的取值范围及允许的操作进行抽象描述	数据类型
多个相同数据类型的一个特征抽象描述	一维数组
多个相同数据类型的多个特征抽象描述	二维数组
多个不同数据类型的多个特征抽象描述	结构体
对系统功能和功能调用的抽象描述	函数的定义和调用
系统	程序或软件

自动化就C语言而言，则是利用该语言的语法规则编写语句，按照顺序自动执行语句并进行自动转换。运用该框架，构建数学模型、设计算法策略和数据结构，通过C语言实现问题工程，学生通过不断的计算思维训练和编程实践，将有助于培养分析设计能力和工程设计能力，从而解决复杂工程问题，达到新工科人才培养的目的。但是C语言程序设计是面向大一新生的课程，理解抽象和自动化培养计算思维是比较困难的。需要在教学案例中针对实际的问题反复采用计算思维方式来分析问题，逐步培养使用C语言描述实际系统的思维。但这一映射过程达到熟能生巧的地步，学生就学会了如何定

义变量，如何选择逻辑结构，如何合理组织算法，如何编程实现客观世界到逻辑世界的映射。如对以下问题的求解，见表2所示。

表2 计算思维求解小数位表

Tab.2 Computational thinking solution	
实际问题	真分数a/b化为小数后，小数点后第n位是多少？
数学描述	$Q = n_1/n_2 = 0, d_1d_2d_3d_4 \cdots d_n, n_1 \in \{1, 2, 3, 4, \cdots\}, n_2 \in \{1, 2, 3, 4, \cdots\}, d \in \{1, 2, 3, 4, \cdots\}$
数据结构抽象	数据存储采用整数类型的变量，分别有分子、分母、小数点后第n位数字
算法描述抽象	
算法策略抽象	第一步：余数*10 第二步：余数*10/除数取整(小数点后n位) 第三步：余数*10%除数求余数 第四步：重复上述过程(映射为循环结构的抽象)
核心代码实现	<pre>int child ;//分子 int mother;//分母 int remainder ;//余数 remainder=child; for(i=1;i<=n;i++) { number=remainder*10/mother; remainder=remainder*10%mother; } </pre>

在教学中贯穿计算思维能力训练，特别是在解决实际问题时展现科学分析问题过程，完成数学抽象，分析如何将数学模型映射为对应的数据结构和符合逻辑的算法设计，展示编程与调试，以及生成可执行的程序的过程，实现对计算思维的诠释。这将保证学生的基础知识、分析设计能力、工程设计能力的达成度。

3.2 基于MOOC完成自构建的过程监督体系

建构主义学习理论提倡在教师指导下的、以学习者为中心的学习，从而培养学生的自我学习能力以适应新工科人才的需要^[8]。MOOC平台是实现建构主义学习的有效平台，通过该平台，突出学习者的认知主体地位，体现教师的指导、帮助和促进作用，教师是整个学习过程中的组织者、帮助者和促进者，实现从传授者和灌输者的转变。角色功能如图4所示。

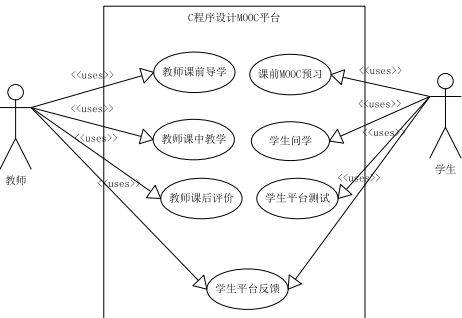


图4 角色功能图

Fig.4 Role functions diagram

在该平台中，教师将C程序设计中的基础语法知识的相关内容发布到平台中，学生根据教师指令，以及自身情况提前在平台中有选择的完成相关基础知识的自主学习，在课堂教学中，根据学生的学习情况针对特定问题进行讲解，节省了大量的课堂时间，教师可充分利用剩下的时间探讨实际问题的解决方案，将有充分的时间进行计算思维、工程分析设计、工程规范、程序分析与调试方面的训练。

同时，平台具有记录学生的学习痕迹的功能，将学生的学习时间、学习效果、学习程度等相关信息以统计的方式进行呈现，教师能及时发现学生的问题，实现快速反馈和分层指导，实现过程监督的有效性。同时将学习过程纳入考试成绩，可有助于提升学生对过程学习的重视程度，保证学习效果，排除期末临时死记硬背应付考试的情况。

3.3 层层递进实现知识内化的工程设计能力

学生在构建C语言知识体系时，可通过MOOC线上平台完成基础知识体系的构建，形成C语言的关键字、规则、规范的识记。在线下，在课堂活动中，通过计算思维训练形成分析、设计、实现实际问题的求解，在求解过程中，教师融入工程设计的一些理念、框架培养工程设计能力。在C程序设计主要思想见表3所示。

表3 工程思想表

Tab.3 Table of software engineering ideas	
思想	作用
命名规范	提供代码可读性、可维护
注释	提供代码可读性、可维护
计算机工作原理	按照输入、处理、输出的逻辑编程
模块化编程	高内聚、低耦合，单一职责便于分工和维护

通过以上两步基本完成学生自我能力和工程设计能力的培养，除此之外，还需培养学生的创新、创造能力，教师在教学设计中，需要构造合适的教学案例实现学生知识内化，通过层层深入完成高阶思维的建立。高阶思维的建立过程实际上是各部分知识的由点、线、面、网的组合过程，实现对问题求解的简单到复杂的，实现解决方案的精益求精。如求解输入一个三位数，求逆序数，例如输入123，则得到321，见表4所示。

表4 高阶思维构建过程表

Tab.4 Process of high-order thinking construction	
逆序数问题	解决方案
给定一个具体的三位数	利用除法和取余原理，采用顺序结构实现
输入任意一个三位数	添加保证是否为三位数的有效性判断
输入任意一个正整数数	使用循环结构判断
输入任意一个正整数数，保证代码的重用	使用函数求解逆序数
添加判断该数是否是回文数的判断	添加回文数判断函数

```

int isBackLetter(int num)//体现数据的任意性
{
    int oldNumber,newNumber,digitNumber;//老数,新数,个位数
    newNumber=0;
    oldNumber=num;
    while(oldNumber!=0)//利用循环结构求解逆序数字
    {
        digitNumber=oldNumber%10;
        newNumber=newNumber*10+digitNumber;
        oldNumber=oldNumber/10;
    }
    if(num==newNumber)//回文数的判断
    {
        return 1;//1表示回文数
    }
    else
    {
        return 0;//表示不是回文数
    }
}

```

程序设计的学习从模仿开始,但在知识学习的积累过程中,要逐渐培养自己的思维逻辑,教师要积极引导进行思维训练,鼓励学生拓展思路,积极探索,采用不同方式实现一题多解,寻求更健壮、更容易维护的算法。实现知识内化到高阶思维的转变。

4 程序设计实践探讨(Practice discussion)

针对C语言程序设计,实践是提升分析问题、解决问题的唯一途径。只有在实践过程中才能发现问题,才能找出自身的不足。在不同阶段开展不同的实践,见表5所示。

表5 程序设计实践阶段图

Tab.5 Stages of programming practice

阶段	目的
秋季学期课程实施阶段完成编程认知实践	基于计算思维,在课程实验环节训练学生对C语言的基本知识运用能力
冬季短学期(两周)算法设计实践	结合二课课外科技活动,开展学科竞赛训练(南桥杯、ACM),该阶段主要培养学生的抽象思维能力、建模能力,问题求解的思路和方法是提升计算思维能力的关键
春季学期(第一周)综合设计实践	基于应用需求,在软件工程思想指导下,进行系统设计和开发。将实现算法、设计的综合应用,也将实现函数、指针、结构体的综合运用,实现到数据结构后置课程的衔接与过度
春季学期(最后一周)综合设计实践	结合数据结构,将工程与算法相结合,实现问题分析设计,算法设计、工程设计的综合能力培养

对于实践的考核,一直以来是一个比较困难的问题,实践过程存在抄袭的问题,如何判断是否抄袭,如何阻止这一现象发生的最好方式就是实施过程监督,过程监督的载体就是实践笔记,笔记中记录实践中遇到的问题,以及解决这类问题的方法,对实践过程进行归纳和总结,并进行反思。每个学生遇到的问题和解决方案是不一样的,容易发现抄袭的学生。同时,实践过程中形成的读书笔记,将是一份宝贵的财富,归纳总结将加速学习的进步与提高。

5 结论(Conclusion)

为满足人工智能引导下的“新工科”人才培养需求,将计算思维融入课程教学中,可使学生发散、模糊、混乱的程序设计思维清晰化、规范化;通过课程的实践环节训练能够有效培养学生的计算思维、建立程序设计编程思想和编程规范和框架;并基于工程思想解决实际问题。并结合MOOC平台提高学生自我学习的能力,并为教师提供了更为充分的时间开展计算思维和内化教学设计,培养学生的分析设计能力和工程能力。

参考文献(References)

- [1] 教育部高教司.新工科建设指南(“北京指南”)[J].高等教育研究,2017(04):20-21.
- [2] 赵广辉.面向新工科的Python程序设计交叉融合案例教学[J].计算机教育,2017(08):23-27.
- [3] 顾丽红,丁淑妍.面向计算思维的蒙特卡罗C语言程序设计案例探究[J].计算机教育,2018(01):105-111.
- [4] 兰丽,胡晓辉.工程认证背景下基于“雨课堂”改进“C语言程序设计”课程关键教学环节[J].计算机时代,2019(01):102-104.
- [5] 季丽琴.以“编、改、调”三部曲改革C语言实践教学[J].计算机教育,2018(04):114-116.
- [6] 蔡美玲,张锦,宾亚玲,等.工程教育理念下《程序设计基础》实验教学改革探索[J].计算机工程与科学,2018,40(S1):21-26.
- [7] 光蓉.融入计算思维的C语言实验教学设计[J].实验室研究与探索,2015,34(10):81-83;103.
- [8] 于波,刘佳,张云晶,等.基于C语言课程的新工科人才编程能力培养支架式教学研究与实践[J].吉林化工学院学报,2018,35(08):28-31.

作者简介:

- 肖 斌(1978—),男,硕士,副教授.研究领域:软件开发.
- 汪 敏(1980—),女,硕士,副教授.研究领域:人工智能,机器学习.
- 王 兵(1977—),男,硕士,副教授.研究领域:软件开发.