

Discussion on the dual-objective teaching method of the general course of “C language programming” in the new engineering subject

Yin Lin Li Gao

Huaide College, Changzhou University, Jingjiang, Jiangsu, 214513, China

Abstract

Under the background of emerging engineering talent cultivation, C Language Programming, as a core general course for freshmen in engineering majors, needs to balance the “ability-value” dual goals: the ability goal focuses on mastering C language knowledge and algorithm design skills, while the value goal aims at cultivating thinking ability, good habits and professional quality. This paper addresses the problems in students’ learning, such as abstract cognitive conflicts, confusion of operation rules, difficulty in thinking transformation, lack of underlying logic, and inappropriate learning methods. It constructs a teaching system of “case-driven, thinking visualization, and value explicitness”, proposes ten teaching methods including case-based teaching, scenario-based explanation, and AI assistance, and clarifies key strategies in teaching implementation such as controlling knowledge granularity and strengthening standardized writing. Practice shows that this system can effectively achieve the dual goals, providing a reference for the teaching optimization of basic engineering courses.

Keywords

Emerging Engineering; C Language Programming; ability-value Dual-goal Teaching

新工科《C语言程序设计》通识课的双目标教学法探讨

林荫 高莉

常州大学怀德学院, 中国·江苏 靖江 214513

摘要

在新工科人才培养背景下,《C语言程序设计》作为工科一年级核心通识课,需兼顾“能力-价值”双目标:聚焦C语言知识掌握与算法设计能力,指向思维培养、习惯养成与职业素养塑造的价值目标。本文针对学生存在的抽象认知冲突、运算规则混淆、思维转换困难、底层逻辑缺失及学习方法不适等问题,构建了“案例驱动-思维可视化-价值显性化”的教学体系,提出案例式教学、场景化讲解、AI辅助等十种教学方法,并明确教学实施中知识点颗粒度把控、规范化书写强化等关键策略。实践表明,该体系可有效达成双目标,为工科基础课程教学优化提供参考。

关键词

新工科; C语言程序设计; 能力-价值双目标教学

1 引言

在普通高校工科人才培养体系中,《C语言程序设计》作为自动控制类相关专业一年级学生的核心通识课程,是学生接触程序设计的入门载体。鉴于多数学生为零编程基础,课程明确新工科“能力-价值”双目标:能力目标聚焦于掌握C语言基础知识、能进行程序编程及计算机问题求解的算法设计;价值目标则指向培养逻辑思维、抽象思维及辩证思维,形成良好自主学习习惯、职业素质与严谨细致的工作

作风,养成科学认知习惯,报效国家,以拓展人生可能。

C语言兼具高级语言与汇编语言特性,可直接操控底层硬件,是后续可编程器件、智能设备开发等系统控制类课程的基础,亦是自动控制领域的核心工作语言,地位举足轻重。然而,C语言功能强大、语法复杂灵活,初学时常因中学知识体系迁移阻碍、思维模式转换等困难,出现语法混淆、误用等问题;即便在人工智能时代,其学习仍对学习者提出较高要求,存在显著难度。

本研究旨在厘清学生学习中的典型问题及成因,提出针对性教学方案,推动课程成为兼顾知识传授、能力培养与价值引领的优质课程,最终达成“能力-价值”双目标。

【课题项目】江苏高校品牌专业建设工程三期项目。

【作者简介】林荫(1970-),女,中国湖北新洲人,硕士,副教授,从事计算机教育,算法与大数据研究。

2 双目标达成的阻碍分析：学习中的典型问题与归因

2.1 C 语言抽象性与人类认知逻辑的冲突，阻碍了逻辑思维与严谨作风培养

C 语言作为机器控制语言，与人类自然语言存在本质差异。人类语言是弹性沟通系统，依赖共识与容错；C 语言是刚性控制工具，依赖绝对严谨与逻辑规则。核心矛盾体现为：

(1) 机器对程序的零容错性与人类认知中模糊共识的冲突，使学生难以适应精确到符号的表达要求，阻碍严谨作风养成。(2) 语义对底层逻辑的依赖性与学生表层记忆习惯的冲突，导致对语法的抽象理解困难，影响逻辑思维建立。

(3) 机械报错的规则导向性与人类“意图推测”认知模式的冲突，增加问题排查难度，削弱学习信心。

抽象性导致的认知冲突，不仅阻碍对语法的掌握（能力目标），更制约逻辑思维与严谨作风的培养（价值目标）。

2.2 数学表达与 C 语言运算规则的形似神异，阻碍了算法设计能力与抽象思维发展

C 语言与数学的符号系统虽形式相似，语义却存在显著差异，形成知识迁移障碍：

(1) 符号功能差异：数学中“=”表示等式关系，C 语言中为赋值操作（如 `if(x=3)` 的误判断）；(2) 数学中“ $2x$ ”表示乘法，C 语言需写为 `2*x`，此类差异导致运算构造频繁出错，直接影响算法实现（能力目标）；(3) 规则隐性差异：数学中 $5/2=2.5$ ，C 语言中整数除法结果为 2（截断小数），需显式转换类型（如 `5.0/2`），学生因忽略规则细节，难以建立“类型约束”的抽象认知（价值目标）；(4) 符号重定义与多重性：如 C 语言中“*”可表示乘法、指针定义或取值，与数学中单一功能的“ \times ”形成认知冲突，阻碍对符号抽象语义的理解。

运算规则的差异不仅影响程序正确性（能力目标），更制约对符号抽象功能的认知（价值目标）^[1]。

2.3 静态描述与动态流程的思维转换困难，阻碍了问题求解能力与辩证思维培养

数学静态结论描述，而 C 语言需动态流程设计，思维转换困难：

流程显式化要求：如数学中 $y=2x^2-1$ 隐含计算步骤，但 C 语言中必须明确“先定义 x 和 y 的类型（`float x,y;`）→给 x 赋值→计算 $y=2*x*x-1$ ；→输出结果”的完整流程，学生需提前定义变量并指定合适的类型，还需根据类型做合适的输入输出格式设置。

步骤完整性要求：数学仅需描述变量关系，C 语言需明确“定义类型→赋值→计算→输出”全流程，学生因缺乏过程意识，难以建立问题分解的辩证思维（价值目标）。

思维模式的冲突既阻碍程序设计能力（能力目标），也制约对“问题解决需分步实施”的辩证认知（价值目标）。

2.4 计算机底层逻辑认知缺失，阻碍了系统思维与科学认知习惯形成

C 语言与硬件的紧密关联，使底层认知缺失成为显著障碍：

(1) 变量类型约束：数学中“ x ”可代表任意数，C 语言中需定义 `int/float`，学生如果将 3.14 赋值给 `int x`，将导致精度丢失，体现对“数据存储机制”的无知，影响程序健壮性（能力目标）；(2) 内存操作不可见性：数组越界（如 `int a[5]; a[5]=10`）、指针误用（如 `*p` 与 `p` 混淆）源于对“内存分配”无概念，难以建立“程序-硬件”关联的系统思维（价值目标）；(3) 编译报错陌生感：将 `printf` 误写为 `print` 导致“未声明函数”报错，学生因无法解读提示陷入困境，阻碍科学排查习惯的养成（价值目标）。

底层认知缺失既影响代码正确性（能力目标），也制约系统思维与科学认知习惯的形成（价值目标）。

2.5 学习方法从被动套用到主动设计的适应不良，阻碍了自主学习与创新思维发展

中学“解题套路”模式与 C 语言“创造性设计”需求存在显著差异：

(1) 语法学习：学生试图记忆 `for(;;)` 格式而非理解“初始化→判断→更新”逻辑，写出 `sum+i;`（应为 `sum+=i`）等“形似神异”的代码，体现对“语法功能”的被动接受，阻碍自主编程能力（能力目标）；(2) 问题解决：数学题“唯一解”与 C 语言“多解性”（如求阶乘可用循环或递归）形成对比，学生因缺乏“调试意识”（如用 `printf` 输出中间结果）难以定位漏洞，制约创新思维（价值目标）；(3) 符号认知：数学符号的抽象性（如“ x ”）与 C 语言符号的具体性（如‘a’对应 ASCII 码 97）冲突，学生难以关联语法与功能，影响知识迁移能力（能力目标）。

学习方法的不适既阻碍编程实践能力（能力目标），也制约自主学习与创新思维（价值目标）。

3 “能力-价值”双目标导向的课堂教学方法设计

(1) 案例式教学。以“C 语言工具性”为核心，将每个知识点嵌入案例：从“Hello, world!”入门，到用数组处理传感器数据、用指针实现设备状态监控，使学生直观感知 C 语言在自动控制领域的应用价值。案例设计遵循“由易到难”梯度，既逐步提升编程与算法设计能力（能力目标），又通过“用知识解决实际问题”的过程，培养职业认同感与严谨务实的工作态度（价值目标）。(2) 解析式读书法。通过“语言转译”明确表达式语义，如将 $y=3*x$ 读作“ $3x$ 的值赋给 y ”， $y=3*x$ 读作“判断 y 与 $3x$ 是否相等”，帮助学生规避符号歧义，强化“精确表达”意识。(3) 标记法读程序：如用红色标注选择结构中可执行语句、区分素数判定程序的核心逻辑与输出控制语句，使程序流程可视化，

培养关注细节的严谨习惯。(4) 场景化讲解。如用“巡考老师检查学生证件”解释嵌套循环(i 为老师、 j 为学生,体现 i 每变一次 j 遍历全量);用“辅导员与班长”类比全局与局部变量,将抽象语法与生活经验关联,降低理解门槛^[2]。(5) 画图法。如通过内存空间图示展示函数参数传递、用递归调用图分析Fib(n)的资源消耗,直观呈现底层逻辑,帮助建立“程序-硬件”关联的系统思维。(6) 反向构造数学表达式。如将函数返回 $1/0/-1$ 表示 $a > b/a == b/a < b$ 转化为分段函数,用数学语言桥接程序逻辑,强化抽象思维。(7) 多方法解决问题。如针对“水仙花数求解”提供“数位提取”“循环分离”“穷举遍历”三种方法,针对“求面积”融合“底乘高”“三角函数”“海伦公式”,引导学生对比优劣,培养“多维度分析问题”的辩证思维与创新意识。(8) 定式法。如针对冒泡排序、素数判定等核心算法,通过“理解逻辑+重复练习”固化基础模块,既提升算法设计熟练度(能力目标),又通过“刻意练习”培养坚持与专注的学习习惯(价值目标)。(9) AI辅助。借助AI定位语法错误(如将print修正为printf),或逐句解析代码功能,辅助学生自主排查问题,提升自主学习能力与科学探究习惯。(10) 价值激励融入。在教学中嵌入“价值引导”:通过分析“数组越界导致设备控制系统崩溃”的案例,强调严谨作风的工程意义;通过分享程序员“用算法优化生产流程”的实例,激发职业责任感;通过小组合作完成“简易温控程序设计”,培养团队协作与问题解决的担当意识,使价值目标从“隐性”转为“显性”^[3]。

上述教学方法形成“知识-能力-价值”三位一体的培养体系。通过案例与定式法夯实编程与算法能力(能力目标),通过场景化与画图法培养逻辑、系统思维(价值目标),通过多方法与AI辅助提升自主学习与创新素养(价值目标),最终实现双目标的协同达成。

4 教学实施中的关键问题:双目标达成的实践保障

4.1 知识点颗粒度的合理把控

需根据一年级学生认知基础,避免过度深入底层细节:用“数据类型取值范围”解释溢出(而非内存位存储),初学指针聚焦“变量地址关联”(而非行指针等复杂应用)。此举既保证能力培养的循序渐进(避免因难度过高挫败学习信心),也符合“由浅入深”的科学认知规律(价值目标)。

4.2 程序规范化书写的强化

对比“压缩式代码”(如if(a)d=d-10;else if(!b)if(!c)d=15;else d=25;)与“规范化代码”(缩进明确、结构清晰)的可读性差异,强调规范化不仅是“格式要求”,更是“逻辑清晰”的体现——既便于排查错误(能力目标),也培养“条理清晰”的职业素养(价值目标)^[4]。

4.3 规避无意义的语法试探

摒弃教材中类似“运行while(i++ < 4);后求i值”这类刻意设计的晦涩表达,聚焦“用程序解决实际问题”的本质。此举可减少“语法技巧”的无效消耗,引导学生关注算法逻辑与工程应用(能力目标),同时传递“务实求真”的学习态度(价值目标)。

4.4 算法设计广度深度的梯度适配

作为入门课程,算法设计需与学时、学生能力匹配:以合适为主,暂不要求ACM竞赛级复杂算法。既保证多数学生能达成能力目标(避免挫败感),也通过“跳一跳够得着”的挑战,培养“循序渐进、持续提升”的成长思维(价值目标)。

教学实施需通过颗粒度把控、规范化强化、内容聚焦与难度适配,为双目标达成提供实践保障:既确保能力培养的可行性,也为价值目标的落地创造适宜环境。

5 研究结论与教学效果反馈

5.1 研究结论

本研究针对《C语言程序设计》“能力-价值”双目标,揭示了学习中的典型问题(抽象性冲突、规则差异、思维转换等)对双目标达成的阻碍机制,构建了“案例驱动-思维可视化-价值显性化”的教学体系,提出了颗粒度把控、规范化强化等实践策略,为课程教学优化提供了系统性方案。

5.2 教学效果反馈

经过多年实践打磨,该教学体系成效显著。

(1) 能力目标达成:学生扎实掌握C语言核心语法,能独立设计常规问题求解算法,完成从自然语言到程序表达的思维转换,为后续专业课程奠定基础。(2) 价值目标落地:学生逻辑思维、抽象思维显著提升,形成“调试排错”的科学认知习惯与“代码规范”的严谨作风,自主学习能力与职业认同感增强,部分学生通过课程激发编程兴趣,拓展了职业发展可能。(3) 教学相长:教师在探索中提升了“知识传授与价值引领融合”的教学能力,形成可持续的教学成长路径,课程满意度持续提升。

综上,该教学体系有效实现了“能力-价值”双目标的协同培养,为工科基础课程的教学优化提供了可推广的实践范式。

参考文献

- [1] 向艳.C语言程序设计(第3版)(M).北京:清华大学出版社.2019.8
- [2] 彭响华,崔晗,彭玲,等.“C语言程序设计”课程教学改进及研究[J].通讯世界,2024,31(11):52-54.
- [3] 周燕,周月霞,曾凡智,等.C语言程序设计课程思政探索[J].计算机教育,2024,(10):82-88. DOI:10.16512/j.cnki.jsjy.2024.10.022.
- [4] 朴明.C语言程序设计课程的教学策略分析[J].集成电路应用,2024,41(10):422-423. DOI:10.19339/j.issn.1674-2583.2024.10.202.