

# 面向能力培养的程序设计基础课程改革实践<sup>\*</sup>

谢晓艳<sup>1</sup> 谢晓巍<sup>2\*\*</sup> 曹伟<sup>2</sup>

1. 湖南大学信息科学与工程学院, 长沙, 410012      2. 长沙理工大学计算机与通信工程学院, 长沙, 410114

**摘要** 程序设计基础课程是高校计算机基础教学的核心课程, 本文针对传统程序设计课程教学中存在的问题, 提出以能力培养为核心, 从教学内容、教学方法、实验教学和考评方式等方面进行改革探索, 以提高教学实效, 使学生在提高程序设计能力的同时, 培养其计算思维能力和解决复杂问题的能力, 在教学实践中取得了较好的效果。

**关键字** 程序设计, 能力培养, 课程改革, 考评方式

## Reform Practice of Basic Course of Programming for Ability Training

XIE Xiao-yan<sup>1</sup> XIE Xiao-wei<sup>2</sup> Cao wei<sup>2,2</sup>

1. College of Computer Science and Electronic Engineering, Hunan University, Changsha 410011, China; csxiexy@163.com  
2. School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha 410114, China; xiesir@qq.com

**Abstract**—The basic course of program design is the core course of computer basic teaching in universities. Aiming at the problems existing in the teaching of traditional program design course, this paper takes ability training as the core, and carries out reform and practice from the aspects of teaching content, teaching methods, experimental teaching and evaluation methods, so as to improve the teaching effectiveness, and has achieved good results in practice.

**Key words**—Programming, Teaching methods, Content of courses, Evaluation method

### 1 引言

程序设计基础课程是计算机基础教学的核心课程。该课程通过学习一门计算机语言, 使学生了解高级程序设计语言的基本结构, 逐步理解和掌握程序设计方法; 通过编程训练, 掌握基本的程序设计过程和技巧, 理解计算思维的思想和方法, 具有一定的分析问题和利用计算机解决问题的能力。

### 2 存在的问题

(1) 程序设计基础课程作为非计算机专业的公共基础课程, 课时少, 涉及的概念庞杂。本人所在的高校, 程序设计课程的理论授课是 40 学时, 课程内容多, 教学任务繁重, 知识点关联性强, 学生难以在较短的时间内通透地学好一门程序设计语言。

(2) 教材重语法, 轻编程实践。教材围绕语法展开知识点, 教学过程过分强调语法, 纠缠于一些小细节, 忽略了学生计算思维能力的培养<sup>[2]</sup>。

(3) 评价手段单一。课程成绩由平时成绩和考试成绩综合评定, 考试题型不外乎填空题、选择题、程序阅读题、编程题等, 只要掌握基本的语法, 即使不会编程, 学生也能考试通过。

(4) 由于地域差异, 学生的计算机基础不同, 差异很大, 有些学生有一定的编程基础, 但有些偏远地区和农村的学生甚至没接触过电脑<sup>[3]</sup>。

### 3 教学改革与实践

针对以上问题, 结合本人在多年的程序设计课程中积累的经验, 在教学过程中进行了一些有效的尝试。

#### 3.1 重构教学内容

教学中, 精心设计教学内容, 关注各知识点之间的联系, 使学生建立系统的、全局的知识架构。

以能力培养为核心, 通过教学进程重构, 循序渐进地组织教学内容。以程序设计为主线, 重点围绕学生程序设计能力的培养, 强调编程实践, 以编程带动对语法知识的理解和掌握。

在教学中, 结合实际案例将程序设计的相关内容渗透到每个知识点的讲授中。教学内容按照: 简单的

<sup>\*</sup>基金资助: 本文得到 2022 年湖南大学教学改革研究项目(2022036)资助

<sup>\*\*</sup>通讯作者: 谢晓巍, 副教授, csxiexy@163.com

C 语言程序—分支结构—循环结构—函数—数组—指针—结构体—指针进阶—文件—位运算。其中，在概论这章中，通过几个简单的 C 语言程序，使学生了解 C 语言程序的基本结构，快速入门，提高学习效率。此外，为了使学⽣尽快掌握程序设计的基本思维方式，将部分难点内容如链表放到课程的后期教学中，避免内容难点干扰学生对程序设计基本思维方法的理解。

表 1 教学内容

教学内容	学时
简单的C语言程序	2
分支结构	2
循环结构	6
函数	8
数组	6
指针	4
结构体	4
指针进阶	4
文件	2
位运算	2

### 3.2 案例驱动

在教学中，采用案例驱动的方法，结合实际案例组织和实施教学，将程序设计的方法贯穿到每一个知识点中。

#### (1) 选用经典案例，循序渐进，逐步深入

采用案例引导，以程序设计为主线，逐步深入，拓展学生的思路，打开思维空间，培养学生的计算思维能力。

教学中，选用一些典型的应用型案例，如：斐波那契数列、素数、求回文数、完数、万年历等，由浅入深、循序渐进地将知识点植入问题的求解过程中，训练学生的逻辑思维和计算思维能力，使学生具有运用计算机知识和工具解决一定复杂度问题的能力。

譬如，以求回文数为例。在学习循环结构时，循环的处理过程是：在循环体内，对该数不断地模 10 取余，并且除以 10 取整，直到为零，即可求到这个数的逆序数，然后，判断这个数和原来输入的数，如果相等，则该数是回文数，否则，不是回文数。在学习函数时，要求使用函数编写回文数程序，熟悉函数的定义、函数的参数和函数的调用形式，如程序 1。在学习数组时，求输入的一串字符是否是回文字符，使用数组的处理过程是：在循环体内，只要判断对应位置的字符是否相等，即第一个字符是否等于最后一个字符，第二个字符是否等于倒数第二个字符，…，则

可判断是否回文数，如程序 2，程序更加简洁，执行效率更高。在学习指针时，使用指针求回文数。

```
#include <stdio.h>
int hw(int t)
{
    int n = 0;
    while(t != 0)
    {
        n = n*10 + t%10;
        t /= 10;
    }
    return n;
}
int main()
{
    int i, m, n;
    scanf("%d", &m);
    n = hw(m);
    if (n == m)
        printf("YES");
    else
        printf("NO");
    return 0;
}
```

程序 1 函数

```
#include <stdio.h>
#include <string.h>
int main()
{
    int i = 0, k, flag = 1;
    char m[10];
    gets(m);
    k = strlen(m);
    while (i < k/2)
    {
        if (m[i] != m[k-i-1])
        {
            flag = 0;
            break;
        }
        i++;
    }
    if (flag == 1)
        printf("YES");
    else
        printf("NO");
    return 0;
}
```

程序 2 数组

#### (2) 通过算法多样性培养创新思维

在程序设计的教学中，可以通过算法的多样性培养学生的创新意识与创新思维。由于学生的知识结构、思维发展和思考角度不同，分析问题的方法存在差异，教师应该尊重个体差异，鼓励学生独立思考，强调问题解决中算法设计的多样性、解决同一问题可用不同的算法，训练学生的逻辑思维和分析问题、解决问题的能力。

鼓励学生用不同算法来求解问题，以促进思维发展的深度和广度，培养学生的计算思维能力和创新意识。比如，对于排序问题，可以采用冒泡、选择、快速、堆排序等多种算法实现，鼓励学生编写不同的程

序来实现,并且对多种排序方式进行比较分析,激发学生的思维活动,培养学生计算思维的多样性。

在教学中,从案例入手,引导学生积极思考,从不同的角度考虑问题,鼓励学生提出好的算法,讲解自己的思路,拓展了他们的思维空间;算法的多样性给了学生探索的空间和动力,有利于培养学生的进取精神和创新精神。

### (3) 优化算法,激发创新思维

在提倡算法多样化的同时,教师还要引导学生对算法进行比较和研究,达到简化并优化算法的目标。

教学中,选择经典案例,拓宽学生的思维空间。例如,“百钱百鸡”问题,可以用“穷举法”来求解,它的基本思想是:列举出所有可能的情况,逐个判断有哪些是符合问题所要求的条件,从而得到问题的全部解答。使用穷举法,对每种可能的公鸡、母鸡和小鸡的数目,判断是否符合百钱和百鸡的条件,用三重循环实现,循环执行  $100 \times 100 \times 100 = 1000000$  次,时间复杂度是  $O(100^3)$ ;一个算法包含的循环嵌套的层数越多,执行效率越低,该算法的时间复杂度越高;启发学生可不可以对算法进行简化,用二重循环实现,鼓励学生设计算法并实现,学生发现使用二重循环,循环执行  $100 \times 100 = 10000$  次,时间复杂度是  $O(100^2)$ ;一个简单的改进,使循环执行的次数大大减少,降幅达到了 99%,使学生感受到了算法改进的魅力,提高了学生的学习积极性,学生们主动思考,提出还可以进一步简化,经过分析,购买公鸡的数目不能超过 20 只,购买母鸡的数目不能超过 33 只,循环次数降到了  $20 \times 33 = 660$  次,比二重循环又减少了 93%,极大地提高了程序执行效率。通过对算法进行优化,充分地调动了学生的学习兴趣,促使学生积极思维,活跃了课堂气氛,提高了教学实效。

## 3.3 利用网络教学平台开展实验教学

(1) 利用编程能力测试平台,根据教学进度分知识点、分章节建立实验相关资源。实验之前,先建立实验内容,明确实验要求;实验发布后在网络平台上让学生反复练习,不断刷题、刷分,要求他们边做边总结,积累经验。通过随练随测不断巩固和提高学生的编程能力和应用计算机解决实际问题的能力,培养学生的计算思维。

在实验中,对于比较难的题目,给出思路,引导学生积极思考<sup>[4]</sup>;对于学生遇到的编译和运行错误,鼓励和帮助学生根据系统提示信息,找出错误位置和原因,分析其是语法还是算法的问题,以便下次避免,不要轻易放弃,及时总结,提高实验课的效率,提高分析解决问题的能力。

每次实验后,及时总结,将实验的多种算法和学生中较好的算法进行共享,开拓学生的思路,提高学生分析解决问题的能力。



图 1 编程能力测试平台

(2) 利用泛雅网络教学平台,补充一些综合性的实验进行强化练习。例如:通讯录、学生学籍管理系统、图书管理系统等。

综合性实验题由教师在学期的中后期发布,学生自主选题,可以一人一题,也可以 2 人或者 3 人一题,学生利用课余时间完成;教师指导学生,把问题划分为一个一个模块,进行模块化,培养学生项目开发的意识;遇到问题,上网查阅相关资料,也可以通过平台的论坛和 QQ 群等及时与老师交流、研讨,获取帮助;学期末,利用实验课的时间,每个小组展示自己的成果。

由于综合实训涉及到所学的全部知识点,因此,通过综合实训,不仅可以提高学生的自学能力,提高其解决实际问题的能力,还能促进其计算思维和创新思维的发展。

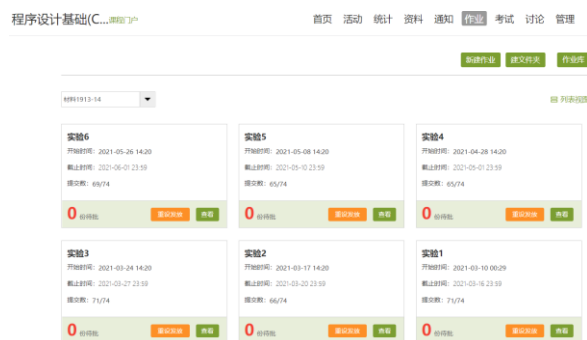


图 2 泛雅网络教学平台实验

## 3.4 多维度的考评体系

强调过程导控的多维度考核评价体系,更加注重过程化考核<sup>[5]</sup>,总评成绩由平时成绩、期中考试、期末考试组成,全部采用机试,随机抽题,随练随考。

新的考评体系:平时成绩一(30%)+平时成绩二(30%)+期末考试(40%)组成。

平时成绩一:5次平时成绩的平均

平时成绩二：3 次平时成绩\*0.2/次+0.4\*期中考试成绩

总评分=0.3\*平时成绩 1+0.3\*平时成绩 2+0.4\*期末考试成绩

平时成绩包括平时作业和测试，平时作业也在系统提交，自动判分；平时测试又包括小测试和章节测试，全部采用机试。上机随测，自动判卷，更能反映学生的真实水平。

上机随测，每月有一次月考，每章有一次考试，自动组卷，全部采用机试，都计入平时成绩，使学生一时都不能松懈，只有不断刷题，才能保证不凋队，得高分。避免了抄作业，考前突击的现象。学生的积极性高，逻辑思维和计算思维能力提高快。

同时，鼓励学生参加 CCPC、ACM 等程序设计竞赛，并在平时成绩中进行加分。

采用新的考核系统，对学生提出了更高的要求，考核内容覆盖面更广，更有利于激发学生的学习积极性和学习兴趣。

## 4 改革成效

通过不断的教学改革和探索，在教学方面取得了一定的成效，所教班级的程序设计基础课程，在期中、期末全校统一组织的非计算机专业机考中，都取得了名列前茅的好成绩。譬如 2020 年所教班级的平均成绩比学校总的平均成绩高 10 分多，2021 年高 8 分多，如表 2 所示。

表 2 期末考试成绩对比

平均成绩	2020春	2121春
全校总平均成绩	46.99	53.1
所教班平均成绩	57.06	61.67

从图 3 各分数段人数所占百分比中，可以看出，所教班级 60 分以下的低分段占 44%低于学校的 54%，而 90 分以上的高分段远高于学校的平均水平。

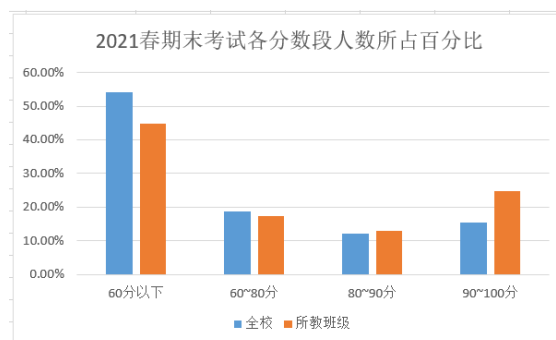


图 3 2021 春期末考试各分数段人数所占百分比

通过调查问卷和学生评教的反馈，学生对课程教学的满意度达到 95%以上，认为课程对程序设计能力的提高有很大的帮助，提高了自身解决实际问题的能力。

## 5 结束语

为提高程序设计基础课程的教学效果，我们从教学内容、教学方法、实验教学和考评方式等全方位进行改革实践，取得了一定的成效，学生学习程序设计的积极性显著提高了，逻辑思维和计算思维能力增强了，后续我们将学习更加先进的教学理念，持续改进，使程序设计课程和学生所学专业更加有效地结合，使学生学有所用，为学生的后续学习打下坚实的基础。

## 参考文献

- [1] 文万志, 程学云等. 面向 OBE 的程序设计基础课程的计算思维培养[J]. 计算机教育, 2020(6): 85-89.
- [2] 丁海燕. 计算机程序设计课程中计算思维的培养[J]. 实验技术与管理, 2017(8): 17-19
- [3] 彭立宏, 肖晓强等. “计算机程序设计”课程教学研究与实践[J]. 计算机工程与科学, 2014(36): 39-42.
- [4] 张泳, 颜晖等. 以应用能力培养为核心的程序设计课程教学全面重构[J]. 计算机教育, 2021(1): 70-74.
- [5] 张广斌, 张润梅等. 混合式教学在程序设计基础课程中的应用[J]. 软件导刊, 2019(18): 176-179.
- [6] 李薇, 黑新宏, 王磊. C 语言程序设计课程的金课建设实践 [J]. 计算机教育, 2020(6): 142-146.
- [7] 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(8): 24-27.
- [8] 李林. 面向综合应用能力培养的 Python 课程教学探索与思考[J]. 计算机教育, 2018(9): 65-68.