

## ◆ 清华基础教育谈

文章编号：1672-5913(2013)17-0112-07

中图分类号：G642



# 教学与教案设计

谌卫军

(清华大学 计算机科学与技术系, 北京 100084)

**摘要：**教书育人是教师的天职，教学工作是教师的立身之本，如何做好教学工作，值得每一位大学教师认真思考。文章介绍笔者在教学与教案设计方面的一些个人体会，以笔者承担教学的计算机程序设计基础和操作系统两门课作示例，阐述课程定位、教学内容设计、教案设计和课堂教学 4 个方面内容，覆盖一门新课从无到有的整个过程。

**关键词：**教学设计；教案设计；计算机基础教学

## 0 引言

对于一名大学教师而言，教学工作是首要工作，是其在学校的立身之本。一方面，很多学校都建立了教学评估机制，由学生评价教师上课水平，而且这种评价将直接影响教师个人的业绩考核和职称评定；另一方面，教师的天职是教书育人，讲好课和培养好学生也是教师自身的职责和内在的愿望。如何做好教学工作，是值得每一名大学教师认真思考的问题。

一般来说，大多数教师虽然在各自的专业方向上是专家，但是在教学上可能并不擅长，尤其是刚入职的青年教师，非常缺乏教学经验。事实上，很少有教师接受过专业的教学培训，对于如何授课，全凭个人的摸索和钻研。笔者也曾经经历过那个阶段，在刚开始从事教学工作时，由于缺乏经验，不知如何上课，基本上是照本宣科、念讲义，结果可想而知。经过数年的学习和钻

研，终有小成，在教学工作上做出了一些成绩，有了一些心得和体会，愿与各位读者分享。

## 1 课程定位

教师在承担一门新课教学，准备具体的授课内容之前，首先要了解该课程在整个课程体系中的定位。例如，该课程是大学几年级的课程，有哪些前置课程和后继课程，每周的授课学时有多少，有无实验课时等。计算机程序设计基础是清华大学计算机科学与技术系为全校本科生开设的一门公共选修课，它没有前置课程，后继课程包括面向对象程序设计、操作系统、软件开发实践等。

其次，教师要深入了解课程的授课对象。所谓“知己知彼，方能百战不殆”，任何一堂课，任何一部分内容，都有难易、深浅之分，而且讲课的速度也很有讲究，对于其中的分寸把握，教

作者简介：谌卫军，男，讲师，研究方向为计算机基础教学、计算机技术和网络学习，cwj@tsinghua.edu.cn。

师只有在充分了解学生背景的情形下才能定夺。另外，学生的基础构成往往不是单一的，既有基础较好的，又有基础较差的，这就增大了授课的难度。

如何了解授课对象的背景呢？一个简单的做法是课堂调查。以计算机程序设计基础课程为例，在第一堂课上，笔者都要进行一个调查，将所有学生按照编程基础分为 4 类，然后进行统计。学生编程基础调查结果如图 1 所示<sup>[1]</sup>。

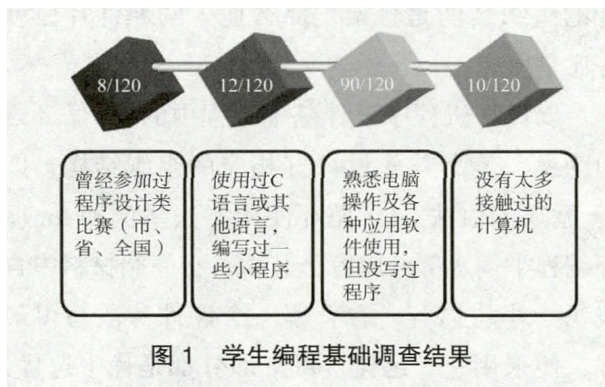


图 1 学生编程基础调查结果

在充分了解课程定位和学生特点之后，我们就要为该课程制订相应的教学理念和教学目标。个人的观点是能力比知识更重要，任何一门课程的教学都应该以提高学生的各种能力为目标，而不仅仅是传授知识。以计算机程序设计基础课程为例，学生的特点是善于学习 C 语言的语法知识，而且对一些琐碎的语法细节很感兴趣，但是在解决实际编程问题时能力明显不足，教师在课堂讲解一些算法设计方面的内容时，学生也表现得很吃力。根据这些特点，我们制定本课程的教学目标，即让学生掌握程序设计的基本原理和方法，提高他们的各种能力，包括自学能力、算法分析和设计能力、动手实践能力和创新能力。

## 2 教学内容设计

每一门课程都有大量内容，而学时数有限，因此什么内容该上课讲，什么内容应由学生自

学，什么内容该重点讲，什么内容应一笔带过，这些都是教学内容设计要解决的问题。

教学内容要围绕教学目标进行设计，内容是为目标服务的。以计算机程序设计基础课程为例，由于目标是为了提高学生的问题分析和算法设计能力，因此教学内容要进行相应调整，少讲知识型内容，多讲能力型内容。在课程安排中，C 语言的语法知识只占课程内容的 1/3 左右，而编程案例则占课程内容的 2/3 左右。这些编程案例并不是一些简单的、传统意义上的例题，即为了演示某个语法（如 for 语句）的使用方法而引入的小例子（代码量只有几行或十几行），而是精心设计的、具有一定难度和工作量的例题，如猜数字游戏、八皇后问题、证券数据文件格式解析等。教师可以通过这些编程案例，向学生讲解问题分析和算法设计的过程。

教学内容设计的一项重要工作是制订教学大纲，每一章、每一堂课讲什么内容，讲多长时间，先讲什么，后讲什么，都要详细列出来。教师在制订教学大纲时要注意，课程内容一定要主次分明，多讲重点和难点，少讲简单的、学生自己就能看明白的内容。另外，课程内容要有创新，教师需根据课程的目标和自己的理解引入新的内容，不能原封不动地照搬他人的教学大纲。

## 3 教案设计

在明确了课程定位，制定好教学大纲之后，接下来的工作就是进行教案设计，这是创建一门新课程的最主要工作。这里主要讨论 3 方面的内容：材料搜集、融会贯通以及教案设计中的一些基本要求和方法。

### 3.1 材料搜集

教案设计的第一步是搜集材料，这是非常重要的一件事情。我们在开设一门新课时，如果

要从头开始一页一页地设计 PPT, 那么会非常困难。毕竟每个人的能力有限, 时间也有限, 如果每一页 PPT 都要亲自从头开始设计, 那么既耗费时间, 效果又不一定很好。教案的制作要“站在巨人的肩膀上”, 搜集尽可能多的相关课程资料, 如课件、参考书和习题等, 这样才能在短时间内制作出高水平的课件。对于笔者而言, 在开设一门新课前, 要专门花一个星期的时间搜集相关材料。材料的来源很多, 一是来自于其他教师, 包括校内和校外承担相同课程教学的教师, 可以向他们索取; 二是互联网, 这是最重要的一个来源渠道, 互联网主要有两个用途, 不仅可以搜索国内外其他大学的相关课程, 而且可以作为教师学习的知识来源, 如我们在学习一个新概念时, 可以到维基百科等网站查询。

如何系统地搜索国内外其他大学, 尤其是国外知名大学的相关课程呢? 笔者的做法是构建一个国际知名大学列表, 然后用搜索引擎搜索每所大学计算机系的主页, 该主页中一般会有一份课程列表及每门课程的描述信息, 笔者通过这些描述信息找到所需课程。例如, <http://www-cs.stanford.edu/courses> 上的内容是斯坦福大学计算机系的课程安排。每门课程都会有一个课程主页, 上面有该课程的所有资料, 包括教学大纲、PPT 课件、作业安排等, 可以对外公开和免费下载。

### 3.2 融会贯通

在材料搜集工作完成后, 教师手上会有很多课件、习题等原始材料, 但对于这些材料不能原封不动地照搬使用, 因为没有任何一份课件是完美的, 而且即使是一份好的课件, 如果不能真正理解它, 那么讲出来也会变味。事实上, 即使是同一份课件, 不同教师讲出来的效果也完全不同。在材料搜集工作完成后要做的事情, 就是归纳整理或深加工这些材料, 将他人的、零散的知识点融会贯通为一个完整的、系统的、自己特有

的课件。笔者的加工方法主要有两种: 东拼西凑法和加工改进法。

所谓东拼西凑法, 即对不同来源的课件片段进行归纳整理, 以使其形成一个有机整体。具体来说, 在讲授某一部分内容时, 教师首先要设计好这部分内容的组织结构, 即分为哪几小块, 每一块的具体内容是什么, 之间的逻辑关系如何; 然后在具体实现时, 针对每一小块的内容选择不同课件。打个比方, 如果教师自己设计的组织结构是骨架, 那么他人的课件片段就好比是血肉。

以计算机程序设计基础课程中的 for 循环语句为例, 笔者首先设计好相应的组织结构, 即 for 循环的语法、for 循环使用方法举例和 for 循环编程举例 3 部分内容; 其次, 在原始材料中查找每一小块内容, 看看哪一个课件片段写得最好, 便采用它。这样, 每个部分都是优中选优, 不仅能够保证质量, 而且使得各个部分之间有了紧密的逻辑联系, 能够有机地融合在一起。

所谓加工改进法, 即该案例最初来自于他人的课件, 题目很好, 但内容写得不是太好, 此时我们可以对它进行加工改造, 使之成为一个更好的例子。例如, 在笔者的课件中, 有一个求 PI 近似值的编程案例, 该案例最早来源于他人的讲义, 但写得过于简单, 只有问题描述和源代码。这显然不符合教学要求, 因此我们对它进行改进。首先是问题分析, 我们将该案例分解为几个小问题, 然后研究其中的规律; 其次是算法设计, 描述解题的基本思路; 第三是编码实现, 如何把算法转换为相应的代码; 最后是运行结果分析, 向学生提出一些思考题。这样就增加了很多新的内容, 事实上, 经过改造之后, 该案例已经面目全非了。

### 3.3 教案设计的要求和方法

有了各种原始素材以后, 我们就可以进行教

案设计和课件制作，以下是教案设计的一些基本要求和办法。

### 1) 组织结构清晰。

第一个问题是组织结构，设计教案要求各章节结构清晰，各部分内容之间前后关联、逻辑性强、一环套一环。

以笔者承担的另一门课程操作系统教学为例，存储管理一章的章节结构如下：单道程序存储管理、分区存储管理、页式存储管理、虚拟存储技术、Linux 的存储管理。所谓存储管理，即把程序装入计算机的内存中运行，如何装呢？最简单的方法就是在内存中只装入一个程序，即单道程序存储管理，但问题是每次只能装一个程序，而现代操作系统要求同时有多个程序运行，因此改进的方法就是分区存储管理，把内存分为若干个块，每个块装一个程序；然而这样又有问题，即分区的大小不好确定，如果太大了会浪费空间，如果太小了又装不下，因此提出页式存储管理，把内存分为很多个格子，然后把每个程序装到若干个格子中。这样还有问题，即程序的大小必须小于内存，否则就装不下。为了解决这个问题，该章又提出虚拟存储管理。最后是一个具体的虚拟存储管理的例子，即 Linux 存储管理。总之，每个章节的内容是前后相关联的，一环扣一环。

另外，对于每个章节的每一部分内容，我们在设计时要有个固定的套路：Why—What—How，即为什么要引入该内容？该内容是什么？如何使用？例如，对于计算机程序设计基础课程的函数一章，我们可先讨论为什么要引入函数，其目的是为了提高工作效率，减少重复劳动；然后讲解什么是函数，C 语言中的函数与数学函数的区别；最后讨论如何使用函数，包括函数的定义、声明和使用。

### 2) 案例教学。

在教案设计中，案例教学是一种非常重要的

教学手段，它将需要讲授的知识融入案例之中。案例教学可以实现具体与抽象、理论与实践的有机结合，从而培养学生分析问题和解决问题的能力。

案例并不是为了演示课程的某一小块内容而引入的简单的小例子，而是精心设计并且具有完整内容的一个整体。以计算机程序设计基础课程为例，一般来说，一个案例需要具备以下 4 个特点：(1) 具有一定规模，考查多个知识点；(2) 具有完整的内容，包括问题分析、算法设计、编码测试、题后讨论等多个环节；(3) 具有一定难度，富有挑战性；(4) 生动有趣，能吸引人。例如，猜数字游戏是笔者课件中的一个案例，该案例可描述为：电脑随机产生一个数字不重复的四位数，由玩家来猜，每猜一次，电脑将显示形如 \*A\*B 的结果，A 代表位置正确数字也正确，B 代表数字正确但位置不正确，总共有 10 次机会。

这个案例来源于真实的游戏，一些手机、PDA 等终端设备上都有该游戏。该案例一方面具有趣味性，学生对游戏都比较感兴趣；另一方面又有一定难度，富有挑战性。从考查的知识点来看，该案例还需要解决以下几个技术问题：(1) 如何随机产生一个数字不重复的四位数；(2) 对于一个四位数，如何把它的每一位数字拆分出来；(3) 对于玩家猜测的一个四位数，如何计算相应 A 和 B 的数量？

在课堂讲解该案例时，我们会从问题分析、算法设计、编码、测试和题后讨论等多个环节逐一细述，针对这一个案例大概需要讲解 35 分钟。

### 3) 启发式教学。

在教案设计中，启发式教学也是非常重要的一种教学手段。事实上，教学的根本目的就是让学生获得知识和提高能力，而要做到这一点，前提条件就是学生的参与；因此，教师在教案设计中，要注重引入启发式教学，提高学生的参与程度。启发式教学的最高境界是对于一个待讲解的

内容,教师尚未细述,而学生已经给出了答案。

以操作系统课程为例,在讲授存储管理一章时,我们不会直接讲述相关内容,而是先提一个问题,即如何实现多道存储管理,然后组织学生讨论。问题描述:给定一块内存空间,需要在里面装入多个程序并运行,在这种情形下,如何进行内存的分配、回收和管理。这种启发式问题的意义在于:由于相关的内容尚未讲解,学生没有任何先验知识,也就没有思维定势。这时,学生就能充分发挥想象力和创造力,进行换位思考,即如果我是操作系统的设计者,该如何解决此问题?这种积极思考的状态非常难得。事实上,在经过思考以后,学生都能有想法,而且这些想法与真正的存储管理方法非常接近,基本上每一种存储管理方法(如分区存储、段式、页式等)都会有学生想到。这样,等到教师真正讲解这些内容时,学生就更容易产生共鸣。

#### 4)教学方法创新。

在教案设计中,任课教师要结合课程的内容和特点,善于发现和总结规律,提出新的适合于该类课程的教学方法。

以程序设计类课程为例,课程的目标是提高学生的程序设计能力。所谓编程,输入是文字形式的问题描述,输出是源代码。整个编程的过程可以分为3个步骤,即问题分析、算法设计和编码。对于每一个步骤,我们可以有针对性地采用不同的教学方法。具体来说,在问题分析环节采用案例教学的方法,让学生通过大量的案例掌握基本问题分析技术;在算法设计环节提出编程模式概念,类似于武术中的“套路”,它是自顶向下程序设计技术的一种具体实现<sup>[2]</sup>;在编码环节,为了提高学生的自学能力和动手实践能力,我们设计并实现了一个功能强大的在线训练系统<sup>[3]</sup>。

总之,针对程序设计类课程的特点,我们将编程的过程分为3个环节,然后对于每个环节采用不同的应对策略,这就是一种教学方法的创新。

#### 5)内容生动有趣。

教案设计的最后一个问题是教学内容一定要生动有趣。一般来说,一次课往往有3小节(至少2小节),每小节45分钟。教师可以换位思考一下,这样长时间坐在座位上,是很容易疲倦的。如果授课的内容比较抽象、枯燥,那就更容易让学生失去兴趣,因此教学内容一定要生动有趣。这样才能吸引学生并给学生容易留下深刻的印象。

如何才能令教学内容生动有趣?方法一是用生活中的例子类比,把单调抽象的内容变成具体形象的内容。例如,在计算机程序设计基础课程中,递归的概念比较难懂,但相信每一位学生都听说过这样一个故事“从前有座山,山上有座庙,庙里有一个老和尚和一个小和尚,老和尚正在给小和尚讲故事。讲的是什么故事呢?他说,从前……”。这个故事正是递归的一个典型例子。

再如,操作系统课程的虚拟存储管理这部分内容比较抽象,不太好理解,笔者首先给学生看卓别林的电影“马戏团”中的一个镜头,一位魔术师在表演大变活人的魔术。有一个柜子,放在一张桌子上,柜子开始是空的,然后魔术师一拉门,走出来一个人;把门关上,再一拉门,又走出来一个人。柜子看上去很小,似乎只能装一个人,但为何装了那么多人呢?谜底是柜子虽小,但它下面的桌子很大。开始时,所有人都躲在桌子下面,然后轮到谁出场,谁就爬到柜子里,这就是大变活人的秘密。虚拟存储管理技术采用的也是类似的思路。柜子好比是内存,内存比较小;桌子好比是硬盘,硬盘比较大。在开始时,整个程序存放在硬盘上,然后要用到哪部分,就把哪部分装入内存,这样小内存就能运行非常大的程序。这就是虚拟存储管理技术的基本思路。

方法二是风趣幽默、寓教于乐。例如,函数这一章有一个知识点:在调用函数时,参数的传

递是单向的值传递，只能由实参传给形参，而形参不能传回给实参。为了让学生更好地理解 and 记住这个知识点，笔者例举如下函数参数传递是单向值传递的程序段。

```
void main()
{
    int Salary, Cars, Houses;
    Salary = 3000;
    Cars = 0;
    Houses = 0;
    DayDreaming(Salary, Cars, Houses);
    printf("%d%d%d",Salary, Cars, Houses);
}
void DayDreaming(int salary, int cars,
int houses)
{
    salary = salary * 3;
    cars += 2;
    houses ++;
}
```

学生看过这个例子后，往往会心一笑并记住了相应知识点。

方法三是图片和动画的使用，所谓一图胜千言。例如，汉诺塔问题是一个经典的递归算法题目，说的是一座寺庙里，有 3 根柱子，有一位僧人想把一根柱子上的 64 个盘子移到另一根柱子上，在移动时必须遵守几条规则。在讲解这个例子时，教师如果用语言或文字描述，会非常繁琐而且说不清楚，但如果通过制作一个动画以演示移动的过程，那么会非常直观。笔者制作的汉诺塔问题动画演示如图 2 所示。

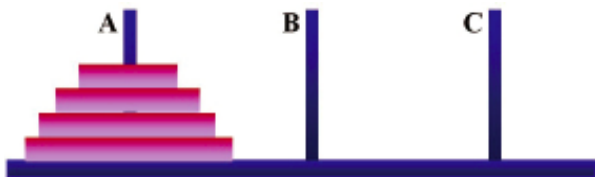


图 2 汉诺塔问题动画演示

## 4 课堂教学

教案设计完成后，下一个环节就是课堂教学，这也是一门课程最重要的环节之一。以下讨论 3 方面的问题：教师的状态、课堂交流以及辅助工具的使用。

### 4.1 教师的状态

在授课时，教师的状态最为重要，直接决定课程讲授的成功与否。如果教师不在状态，那么再精心设计的教案也无用武之地。

教师首先要声音响亮。一方面，从实用的角度出发，声音要足够响亮，让所有人都能听见；另一方面，更重要的是要有一种气势，让学生觉得教师很投入、很有激情，而这种激情会感染学生，使他们也兴奋起来，从而提高课堂教学的效率。

教师除了声音要响亮，还要有适当的表情和肢体动作。在交流过程中，教师的表情和肢体动作往往比言语内容更能传递信息，更能吸引学生的注意力。另外，教师要处理好授课内容与 PPT 之间的关系。作为一名教师，最忌讳的一种上课方式就是照本宣科式地念 PPT，这种方式很容易引起学生的反感，从而导致课程教学的失败。

### 4.2 课堂交流

在授课时，课堂交流也非常重要。任何一堂课都应该以学生为中心，师生互动要充分。课堂交流有两个目的：首先，调动学生参与课堂的积极性，毕竟教学的根本目的是让学生获得知识，如果学生不参与或者消极应付，那么就失去了课程教学的意义；其次，帮助教师了解学生的当前状态，学生是否已听懂，听懂了多少，然后教师根据这个状态调整授课进度。

课堂交流的形式包括 3 种：教师提问学生回答、学生提问教师回答、学生提问学生回答。在

教案设计和课堂教学安排中,教师要有意识地设计和使用这些方式,让课堂交流更加充分。例如,在讲授很多知识点时,对于一些小问题,笔者会故意将答案从 PPT 中抠掉,然后先让学生自己思考,看他们能否做对,通过这种方式了解学生是否掌握了这部分内容。

有时笔者还会在课件中设计一些小陷阱,如在计算机程序设计基础课程中,在讲解 if 语句的用法时,笔者这样设计教案:首先介绍 if 语句的语法“if (E) S;”,其含义是如果表达式 E 的值为真,则执行语句 S,否则什么也不做;随后笔者会给出如下一个 if 语句例子。

```
if (temperature > 38)
printf("你发烧了\n");
printf("速去医院\n");
```

每当讲到此处,笔者会故意停顿一下,然后观察学生的反应。如果学生确实在认真听讲,那么必定会有人站起来,指出该例子是错误的。

#### 4.3 辅助工具的使用

教师在课堂教学过程中,要根据所授课程的特点和内容,选择合适的辅助工具,目的是以直观形象的形式展示相应内容。事实上,如果教师站在讲台上,滔滔不绝地讲上很长一段时间,那么不管他讲得有多么精彩,学生也容易困倦,尤其对于理工类课程更是如此。如果每隔一段时间,教师能够进行实物演示,那么效果会更好。

例如,在计算机程序设计基础课程中介绍基于分治策略的递归算法时,我们先用文字形式介绍这部分内容,但这样给学生的印象不深;随后我们便拿出一个道具,即俄罗斯套娃。所谓俄罗斯套娃,即 6 个外形完全相同但大小不一的洋娃娃,小的装在大的里面。俄罗斯套娃与基于分治策略的递归算法基本原理类似,有了这样一个道具以后,学生对基于分治策略的递归算法印象就更深刻了。

此外,程序设计类课程经常需要编写程序。如果这些程序是以静态的文本形式放在 PPT 中,那么就有些单调和枯燥;因此笔者在课堂上经常使用的一个辅助工具就是 Visual Studio 编程工具。对于课件中的编程例题,我们经常把代码从 PPT 中摘出来放在 Visual Studio 中,进行现场编译、链接、调试和运行,随时有问题随时改进,这样课堂氛围会更好。

#### 5 结 语

课程定位要准确,教师既要了解课程,又要了解学生,然后制订先进的教学理念和教学目标;教学内容的安排要合理,重点突出、主次分明并且有创新点;教案设计要追求完美,不仅要参考借鉴,还要融会贯通,形成自己的风格。在进行课堂教学时,教师需充满激情,多与学生交流,才能活跃课堂气氛。总之,教学是一门艺术,艺无涯,则学无涯。

#### 参考文献:

- [1] Chen W J, Li X, Liu W D. Teaching computer programming to non-computer science students[C]//Proceedings of the 3rd Asian Conference on Education (ACE). Katakira: IAFOR Publications, 2011: 784-795.
- [2] Chen W J, Li X, Liu W D. Teaching computer programming courses using programming patterns[C]// Proceedings of the 2012 3rd International Conference on E-Business and E-Government. Washington DC: IEEE Computer Society, 2012:606-609.
- [3] Shi K L, Chen W J, Zhang L, et al. Kaleidia: A practical e-learning platform for computer programming courses[C]//Proceedings of the Canada International Conference on Education (CICE). Toronto: University of Toronto, 2012: 103-108.

(编辑:宋文婷)