**面向对象程序设计的教学实践与思考**

段学庆

（江苏大学京江学院 江苏镇江 212013）

**摘要**：面向对象程序设计方法尽量模拟、映射人类的思维方式，尽可能使软件开发方法与过程接近人类认识世界、解决现实问题的方法和过程，也就是使得描述问题的问题空间与解决方案空间在结构上尽可能一致，把客观世界中的实体抽象为问题域中的对象。学生在学习面向对象程序设计课程时会遇到各种困难，本文从帮助教师快速了解新生状况的角度提出几点教学实践与思考。

**关键词**：面向对象；程序设计；教学实践；教学反思

不同于传统程序设计，面向对象程序设计是一种全新的程序设计范型，它降低了软件设计的复杂性，改善了软件的重用性和维护性，提高了软件的生产效率，是程序设计方法学的一场实质性的革命[1]。许多高等院校将面向对象程序设计及面向对象技术列入教学计划，作为必修课或选修课。本人作为高等院校专职授课教师，在教学与科研过程中发现学生在面向对象的学习过程中遇到诸多困难，因此本文欲通过笔者自身多年的经验总结，以过来人和亲身经历者的角度抛砖引玉，为刚刚踏入教学岗位的新教师提供些微参考，使其快速了解学生及教学特性，缩短适应过程，及时制定或调整对策，共同为提高学生能力贡献微薄之力。

**一、目前的教学状况**

每年的九月份，高校都会迎来新一届大学生。这些莘莘学子经过层层选拔，历经千辛万苦，现在抱着美好的憧憬和家族的期望来到高等学府求学。相信他们一定下决心不辜负大学美好时光，活出不一样的精彩人生，在自己心仪的专业上进深成长。但他们大都对自己所学专业懵懵懂懂，怀着不切实际的幻想，面对实际专业课的学习时往往感受落差较大。有的学生对本专业毕业去向和具备的能力茫然失措，“晦涩难懂”的专业课像拦路虎一样挡住了当初的热情，有的同学加之自卑、内向，背负太多的家庭压力和责任，没有合适的倾诉对象又不主动寻求帮助，这些因素导致学生在深深的无力感、抑郁、自责、内疚中度过。有的学生在交际、放纵中荒废了学业，落到玩物丧志的地步，着实让人感到惋惜。

除了专职辅导员、心理辅导师及时介入外，任课老师对学生的学习状况最清楚。学生在课堂上表现，或积极参与，或沉默寡言，或若有所思，或疑惑不解，或欲言又止，或畅所欲言，或漠不关心，任课教师都看得一清二楚。笔者现从教师如何帮助学生掌握专业课（尤其面向对象程序设计）角度提几点反思。

**二、教学实践与反思**

**1.引导学生准确定位，增强个体危机意识**

新入校的大学生正经历身份认同上的深刻转变，逐渐由被照料、监管的“未成年人”向在法律意义、自食其力的“成年人”过渡。故此，他们自身的学习模式也被期待有相应变化，从高中时期“填鸭式”的灌输学习，逐步转为在教师引导下的自主学习；以靠记忆背诵为主的题海战术，逐步变为以任务为导向的探究式学习；以强调卷面答题高分的技巧型模式，逐渐改为以提高能力为目标的理论-实践型模式；以凭借自身功底游刃有余的“独行侠”，逐渐被取长补短的团体协作替换。这一系列身心灵全方位的改变，时常会使学生无所适从，找不到人生方向和学习动力，进而想要在象牙塔中补偿高中阶段所受的“苦难”，想要在既有成就的温床上躺平。因此，需要及时引导学生认清现实，不要盲目乐观，摒弃无所谓和玩世不恭的处世态度，了解当前严峻的就业形势，避免陷入“毕业即失业”的窘境。为适应社会大环境，只有提高自身实力，增强危机意识，才能在未来激烈的竞争中占有一席之地[2]。

**2.将士拥趁手兵器，英雄有用武之地**

在明确了目标后，就需要甩开膀子大干一场。“工欲善其事，必先利其器。”作为IT专业的学生，拥有一台性能强劲、携带方便的笔记本电脑似乎已是“标配”。随着人们生活水平的提高，购买一台心仪的电脑并不是高不可攀的奢侈行为。没有电脑，就不能自由自在地用实践验证课堂所学内容，进而体会不到成功的喜悦和经过自身努力排除疑难杂症所带来的欢乐，学生只能是“纸上谈兵”，即使看得懂程序，但因为没有转化为计算机能识别的代码，无法真正检验自己的真实水平。当然，并不是所有的家庭一开始都能负担得起一台电脑的费用，那就需要学校和学生想方设法减少由此带来的损失，比如通过开设公共机房，鼓励勤工俭学等。之所以要强调拥有自己的电脑，另一个原因也是为学生毕业后进入工作着想。试想在IT行业中，有哪个公司是把程序写在纸上就算大功告成的呢？学生若不提早入手熟悉电脑的基本操作和本行业编程软件，毕业后只能眼高手低，无法将智力成果转化为生产力。当然，电脑只是一个工具而已，学校需要积极引导学生自律，避免发生玩物丧志的悲剧。

**3.提升学生自信，培养坚韧品质**

说实话，计算机专业的课程并不容易学，需要花费大量的时间和精力，才能掌握复杂抽象的概念并灵活运用。虽然天赋在整个学习过程中占据一席之地，但更重要的是后天的努力和坚持。因此，要帮助学生克服畏难情绪，树立成功的信心，以IT行业成功逆袭的案例和身边人的实际经历[3]，让其看到“只要功夫深铁杵磨成针”的真实写照，同时打破IT行业的神秘感，不要让晦涩难懂专业名词及其英文缩写成为拦路虎。要坚信，光彩耀眼的专家也是从“菜鸟”开始的，拥有各种响亮头衔的大咖亦是从默默无闻的“草根”起家的，纵横捭阖的互联网大佬也曾经是一无所知的门外汉。就像一首歌中所唱的：“没有人能随随便便成功”。奋斗者最美丽，追梦者最动人，坚持者最精彩。只要紧跟学校开设的课程和老师布置的任务，一步一个脚印，踏踏实实地埋头做，每个人都能干出一番“事业”，成就属于自己的“成功”。

**4.抽象概念生活化，复杂问题场景化**

除了坚定必胜的信心外，采用的形象化讲授方法也会起到事半功倍的效果。刚接触计算机编程的学生往往对抽象、晦涩的专业术语望而生畏，由于没有实感，即使老师讲得唾沫横飞，学生还是云里雾里。这时就需要将抽象概念与学生的实际生活场景联系起来，采用类比、对照的方法和接地气的形式形象地展示出来。以面向对象中的“多态”为例[4]，它包括编译时多态和运行时多态。编译时多态可称为“静态多态”，即在编译和连接时就确定了要执行的程序代码；而运行时多态可称为“动态多态”，就是在编译、连接过程中无法确定所要执行的程序代码，只能等待真正执行的那一刻才确定。此时可以采用学生熟悉的生活场景的一个片段——购火车票从A城市到B城市旅游——进行讲解。静态多态的意思就是在真正出发之前已有一个完整的蓝图：在哪些城市周转，何时进出，会见哪些朋友……这些信息都是确定的，并且不会因为外在因素而改变预定行程；动态多态则没有这些“限制”，可以随着旅行者的喜好和意愿自行决定上述信息，只有在真正购买车票的那一刻，其行为才是确定的。通过这些习以为常的生活类比，学生就比较容易理解多态的概念。

**5.认清现象知其然，把握本质识所然**

千里之行始于足下，不积跬步无以至千里，要想真正掌握计算机编程，是需要长时间的积累，能耐得住性子钻研基础知识，因为这些是构建其他高层建筑的基石和材料。对于每一个概念，要从其内涵和外延、表象和本质等方面进行阐述。对于计算机编程来说，在CPU执行的程序代码与其存储仓库（物理内存）间的对应关系至关重要，沿着数据处理的流程观察内部二进制的变化，便可知晓现象背后的原因[5]。功能与结构，表象与本质，它们之间存在着密切的联系，因此教师需要逐步揭示出隐藏的认知机制，不仅需要帮助学生认识现象，更需要掌握背后的原理、机制，做到知其然，更知其所以然。生活中需要解决的问题的表现形式千变万化，但万变不离其宗，只有掌握背后运行的原理，才不会受限于某一个或某一类型的题目，真正做到举一反三，触类旁通。

**6.加强互动交流，破解课堂沉闷**

教师在课堂上应及时主动关注学生的状态：是疑惑，是兴奋，是烦躁，抑或无奈？通过生动有趣的动画演示，现实场景的再现模拟，让学生参与其中，并在其中发现问题。一旦学生有了切实的体验，知识将不再是冰冷和僵硬的，而是活生生、有温度、“可触摸可感知”的实在[6]。具体形式在这里不一而足，现举几例仅供参考：教师通过分享自己在工作、教学中遇到的实际问题，或者公开匿名演示某一学生的编程作业（类似“大家一起来找茬”游戏），或几人组成临时小组共同探讨解决方案，并让其中一位代表发言，评比最优的算法和思路，以此调动学生积极性，活跃课堂气氛，激发参与热情。这样就在寓教于乐中完成知识的传递与巩固，学生既可以因在预先设置的“错误”陷阱中未发现问题而陷入沉思，体验“一招错,满盘输”的惨烈，此时获得的真知必定印象深刻；也可以在老师的引导、团队的协作下完成题目训练，体验共同体的智慧给整体带来的益处，培养合作、包容、谦虚的精神。实践证明，这样的形式比单纯地点名某一学生站起来回答问题效果更佳，课堂沉闷一扫而空。

**7.引导学生以小见大，实际问题化繁为简**

 刚接触面向对象编程时，学生们面临的问题不会太复杂，一般都是训练基本概念的掌握情况及认真仔细的程度。尽管采用简单的方法对解决某一特定的问题可能更便捷，但这些小题目中所蕴含的原则和思路，却是可以应用在现实生活中的复杂场景中，因此不能小看这些简单题目[7]。例如，比较3个数的大小，可以分别定义3个不同变量来存储它们，它们之间进行两两比较，得出最终结果。在做这类题目时，要想到实际生活的场景，如全校2万名学生按照总成绩排序，此时若创建2万个变量是不可想象的。这样就把数组的概念、快速排序方法引进来。相反，实际问题的描述可能很复杂，乍看上去似乎无从下手，捋不出思路，这时就要抓住问题的本质与核心，剥开那些掩盖着的杂草，利用已学过的知识去复盘它，用已知的模型去试验和校正它。这样学生不会因课堂布置的题目小而胡乱地应付，也不会因实际面临的问题大而恐慌失措，以正确的态度去面对。

**8.学科竞赛阔眼界，比武赛场见高低**

学科竞赛是超出书本范围的一种特殊的竞赛，是强化及提升学生综合能力的独特方式，有助于培养学生解决实际问题的逻辑思维能力，更好地发现和培养创新型人才[8]。学生通过参与学科竞赛，开阔了眼界，增长了见识，结交了朋友，提升自己素质能力，增加以后就业能力的筹码。在竞赛中会遇到来自全国各地的高手，他们独特的思考过程和巧妙的解题思路让人惊叹；在比赛场上更能发现自己的不足和差距，修正自己的目标，改善学习方法，勉励自己奋发直追。鼓励学生跳出自己的舒适圈，勿作井底之蛙，摆正心态面对挑战，既不狂妄自大，认为自己已经“天下无敌”了，也不妄自菲薄，把自己看得一无是处，应该把自己看得合乎中道，正确地评价和看待自己。只要尽力而为，就无怨无悔。人最重要的就是有自知之明，知道自己懂什么，不懂什么，以及如何补足不懂的层面，这才是真正的智者。

**9.培养良好编程习惯，注重文档资料积累**

IT专业的学生一开始可能有个误区，那就是只有编写实实在在的程序，享受指尖在键盘上跳动的快感，那才是程序员的魅力所在；只有像影视作品中的“黑客”练就一身盲打的技艺，不看任何参考资料即可完成任务，看着代码在屏幕上飞舞，那才是程序员的真正价值。现实生活中很多项目都是团队搭配完成的，很少只由一人单独完成大型项目。因此，团队成员之间的交流、互通、分享、协作就显得至关重要。若成员任意按照自己的性子来处理代码，未遵行相应的规范，那么其他开发人员阅读起来将会非常吃力，这样既增加了交流成本，也使得后续的维护也几乎不可能。学生初次接触面向对象程序设计时，就要养成良好的编程习惯，注重文档资料的积累。很多项目由于赶工期，未能遵照程序开发的规范，只注重软件成品的功能，却未能及时记录当初的思考过程和处理结果，进而导致文档空缺或敷衍了事。

文档是IT公司的无形资产，是团队以外的其他开发人员参与项目的最重要的资料。想想看，让一个完全没有接触过该项目的开发人员阅读杂乱无章且无任何文档的100万行代码，将是多么恐怖的事！那又该是多大的工作量！因此，教师在教学过程中要重点强调软件文档的重要性，在软件开发过程中及时、规范地记录各项资料，这比后来再返工将节省大量的人力、物力。

以上就是笔者在讲授面向对象程序设计时所面临的问题及相应的教学反思，虽然带有很多主观感受，但相信也并非个例，仍具有一定的代表性，希望能对从事教学的老师有所帮助。

**参考文献：**

[1] 马金霞,宋传磊.面向过程和面向对象程序设计分析[J].中国新通信,2016,18(11):160.

[2] 白青海.面向对象程序设计基本原则探析[J].内蒙古民族大学学报(自然科学版),2018,第33卷,第6期.

[3] 袁运浩,葛桂萍,李云,等.面向对象程序设计的深度教学模式改革[J].教育教学论坛,2017,第18期.

[4] 陈维兴，林小茶.C++面向对象程序设计教程[M].北京：清华大学出版社，2016.

[5] 蔺想红,郑鉴洋.基于案例驱动的面向对象技术课程教学改革与实践[J].软件,2017,第38卷,第11期.

[6] 李慧.面向电子信息大类的新生研讨课的教学思考与初探[J].教育现代化,2021,第8卷,第49期.

[7] 钱能.C++程序设计教程[M].北京：清华大学出版社，2005.

[8] 杨欣.教育治理数字化转型的利弊及其调适[J].中国电化教育,2022,第11期.

**Abstract**: The object-oriented programming method aims to simulate and map human thinking patterns as much as possible, making software development methods and processes as close as possible to human understanding of the world and methods and processes for solving real problems. This means that the problem space and solution space describing the problem are structurally consistent, abstracting entities in the objective world into objects in the problem domain. Students may encounter various difficulties when learning object-oriented programming courses. The author proposes several teaching practices and reflections from the perspective of helping teachers quickly understand the situation of new students.

**Keywords**: Object-oriented; Programming; Teaching practice; Teaching reflection

作者简介：段学庆（1980-），男，山东临淄人，讲师，硕士,主要研究方向：计算机图形学与GIS。