

## 全国高校混合式教学设计创新大赛教学设计表

(注：表中不要出现教师个人信息、学校信息)

一、课程基本信息			
课程名称	计算机程序设计基础	面向专业	机械，工物，热能，精仪，航院等工科院系
课程性质	√ 必修 ○ 选修	课程分类	○ 通识课 √ 公共基础课 ○ 专业课
学时	总学时：80 线上学时：32 线下学时：48(讲授 32，实验 16)	每学期学生人数	600 人左右 4-5 个平行班
二、课程教学设计方案（整门课程的教学设计）			
1.学情分析与课程目标	<p>（结合本校办学定位、学生情况、专业人才培养要求，具体描述学习本课程后应该达到的知识、能力水平）</p> <p>本课程是理工科非计算机专业的必修的第一门计算机编程能力培养课程，每年有来自机械，工物，热能，精仪，航院等工科院系的大一同学约 600 名学生选修该课程，多个平行课堂，每个课堂 100-150 人左右。</p> <p>该课程的教学目标定位为“零基础开始学习 C++，重实践锻炼计算思维”，使学生掌握程序设计的原理和语法，对后续参与科研或工程进行铺垫，培养学生用计算机解决实际问题的能力，积极推进学生实践能力和创新意识的培养。</p> <p>1.1 学情分析</p> <p>本课程选课面向非电类专业大一新生，主要特点包括：</p> <ol style="list-style-type: none"> <li>1) 具有很强的学习力，但普遍面临着从高中阶段的被动学习方式到知识拓展能力进阶学习模式的转变，学生普遍面临着从高中阶段的刷题学习方式到知识理解拓展和能力进阶的学习模式的转变；</li> <li>2) 学生对知识渴望，但面对繁杂的语法和程序逻辑无从下手，“学了语法搞不定编程；学了概念不知怎么用！”。</li> <li>3) 能力基础参差不齐，学习需求多种多样。以 2021 年春季课堂 3 调研结果为例，29% 的学生之前学过程序设计语言，其中有 3 名同学有信息学竞赛学习经历，71% 的学生没有接触过编程。学习习惯的调查发现也存在多样需求。</li> </ol> <p>1.2 教学目标</p> <p>本课程以 C++ 编程语言内容为主介绍面向对象编程设计方法，秉承“价值塑造，能力培养，知识传授”三位一体的育人理念，本课程的教学目标包括：</p> <ol style="list-style-type: none"> <li>1) 掌握 C++ 语言基础语法，熟悉编程环境，程序设计流程并能完成程序逻辑代码的编写。</li> <li>2) 理解并掌握面向对象的设计方法，包括类的封装，继承，多态，以及模板等面向对象设计模式。</li> <li>3) 具备将逻辑描述转化为代码实现的能力，并能初步分析程序的性能效率。</li> <li>4) 具有一定计算思维能力，培养其工程开发的意识。</li> <li>5) 结合工科专业需求，鼓励创新能力和专业志趣的培养，勇于创新，乐于合作。</li> </ol>		

(本课程教学改革重点解决的问题,混合式教学设计,课程内容与资源的建设及应用情况,教学活动的组织及实施情况,课程成绩评定方式,课程评价及改革成效等)

## 2.1 改革重点解决问题

近年来混合式教学日臻成熟,但面向大班教学的课程如何开展有效的混合式教学仍然需要创新性的设计,具体的难点包括:

- 1) 资源经过多年建设非常丰富完善,但资源的堆砌就能让学生学得好吗?
- 2) 混合的方式和工具平台多种多样,如何做到混而不杂?
- 3) 面对大班上百人不同的需求该如何面对?
- 4) 混合式教学中学习任务重,学生普遍心理没底,混合式教学怎样能不增加学生的学习负担,而达到更好的效果,实现更高阶的育人目标?

## 2.2 混合式教学设计

传统的混合式教学会实施一个线性的连接,将不同的任务串联在一起,但这样的设计并不能从根本上解决以上问题。面向本课程学生的特点,我们进一步梳理混合式教学的资源和模式,设计实施了以 Project 为主线将课前(Pre-class),视频资源(MOOC),课堂(Class)以及在线程序系统(OJ)在内的五大元素高度集成的体系,简称 **P-PMCO 混合式教学设计**,从而打破了传统混合式教学线性的串行关系,资源高效融合,平台物理集成,并在教学设计中形成层次式多角度的混合式模式,层层递进达到高阶育人目标。

1) **课前课中课后的非线性有机融合**: 课前目标导引,各取所需,提供多样路径适应不同需求。课前内容的引导,可以很容易定位知识的漏洞,自主学习视频查漏补缺,课前也作为重要的互动渠道,充分获取学生的需求,并且在课堂上有的放矢交相呼应,在课堂结束的时候给出明确的视频和知识点对应关系,而 MOOC,课后练习,OJ 作为一个整体,在课后起到了巩固提高,真刀实枪练习的效果。

## 2.课程教学设计思路



2) **教学平台高度集成,一键可达**。传统混合式教学任务多,平台多,账号多,我们创新性的将自主研发的 OJ 平台与雨课堂相集成,最终所有的混合式教学流程包括课前学习,课堂互动练习,课程直播回放,课后练习,课件发布,在线编程评测都可以在雨课堂一个平台完成,节奏清晰,混而不杂,对学生来说可以一键可达,而教师后台可以随时监控教学过程实时数据。

3) **混合式集成化的设计不仅仅体现在混合模式的集成和教学工具平台的集成,更重要的是以学生发展为中心的教学过程的一贯性**。结合布鲁姆的教学目标分类,在教学设计上创新性的设计以大作业为核心,以软件工程管理为主线的层次递进式教学模式,通过阶段性教学设计和多种教学模式的融合。我们在整个学期的教学安排上以大作业为主线,从第六周开始布置选题,学生茫然不知所措的时候我们举办在线说明会,让学生看到学到的知识已经可以应用,再提交

选题报告，迭代反馈反复分析评价，并召开在线分享会，分享个体学习的收获，彼此取长补短，朋辈学习，最终映射到每个同学的大作业上。最终让学生从“觉得什么都不会”到“大开眼界，原来还可以这样”最终通过实践取得相应成果。“我原来也可以做到，还想做得更好”！

### 2.3 教学内容和资源建设

教学内容包括了 C++语言的基本语法、面向对象的程序设计方法，以计算思维贯穿始终。自 2014 年开始在全国率先开展程序设计基础混合式教学资源建设，形成了具有示范性的资源体系，具体包括：

- 1) **多层次的视频资源：**课程团队在国内率先建设了程序设计慕课课程，2017 年被评为“国家精品在线开放课程”，2019 年被评为“线上一流本科课程”。并在此基础上不断完善视频资源，目前共建设有超过 40 小时的慕课视频、直播授课和习题辅导视频。与 MOOC 资源相结合的《C++语言程序设计（第 5 版）》82 万字的教材目前已经在全国 300 多所院校使用，该教材 2021 年获得国家级教材建设优秀教材一等奖。
- 2) **集成平台和题库建设：**程序自动评测平台(OnlineJudge)与雨课堂无缝对接，并建设开发了 300 道编程题的在线评测题库，全部集成于雨课堂上便于推广和维护。
- 3) **示范性教学资源建设：**建设有包括 700 页雨课堂课件、并形成了 82 页（超过 10 万字）混合式教学教师手册（含教学设计和翻转课堂讨论题及参考解答），设计超过 100 道雨课堂练习题，可灵活定制穿插于课前或课后。混合式教学的流程和资源可推广并灵活组合，随教材在全国多所院校推广。

### 2.4 教学活动组织实施和评价

经过多年的建设，已经形成了适应本校学生特点的混合式教学模式，具体的教学活动如下图所示：



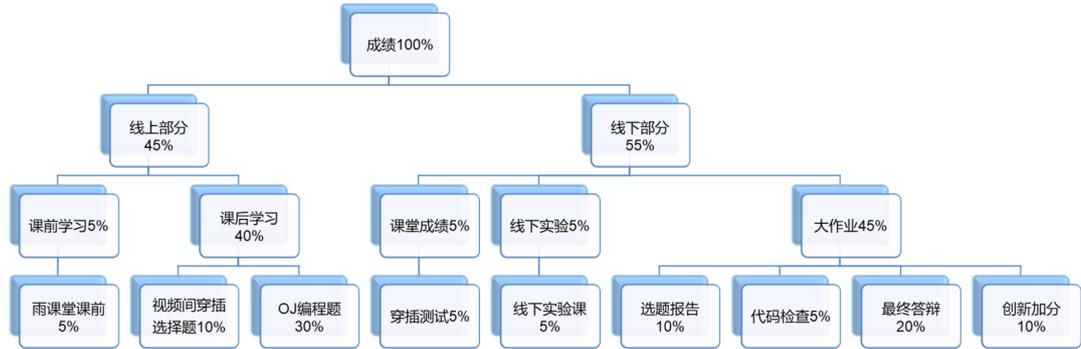
针对面向对象设计中的概念普遍较为抽象而无法短时间内接受的问题，借助雨课堂课前课件推送使学生在课前有所思考，课前提出疑问，课堂上我们针对学生问题进行重点解读，锻炼了学生的批判性思维。针对语法知识枯燥，一看就会一编就不对的问题，我们通过课堂上“挖坑式”的教学模式，抓住易错点设计相应的题目，鼓励学生现场编程验证，小组共同讨论，打破了灌输式的教学模式，提高了学习效果。针对知识点繁琐无法一一展开的情况，我们秉承“不该讲的不讲”的理念，讲授如何搜索关键字，提供相应视频资源，配套练习题，锻炼学生从关键字出发的搜索和学习的能力。因此在本课程中始终致力于对知识点个性的研究和面向学习者的逐步优化，从而以适当的教学设计在适当的时间点教授适当的知识。

每章都有适量的在线综合编程作业（主观综合题）、在线实验指导、在线综合案例解析。期末考核以自选题目大作业的形式，每位同学都会编写千行左右的大作业代码，设计了完整的选题与需求分析、系统设计、编程、测试和答辩等环节，不仅使学生的编程能力得到了强化训练，而且对软件开发的全过程有了初步了解，通过挑战性课题得到全面的锻炼，真正做到了学完这

门课就会写程序，达到了程序设计课的根本目的。

本课程评价体系多元化，结合教学的全流程进行过程性的评价，具体的分值比例如下(线上学习部分占 45%)。

1. 平时线上学习成绩，含课堂练习（检测预习复习和当堂学习情况）、课外在线自动测评的作业 45%
2. 线下实验课 5%
3. 课堂成绩 5%
4. 线下大作业 45%



(用数据或材料说明混合教学的效果，描述课程设计的新颖独特之处及供借鉴和推广的价值)

本课程自 2014 年开始率先进行混合式教学改革，目前探索出适合程序设计基础课程的模式，**创新性的提出了 P-PMCO 模式，改变以往以知识为主线的教学模式，而提出以大作业设计为主线，资源、课堂和在线评测 OJ 相结合的层次式递进模式**，面向混合式教学的难点和痛点开展教学创新，从学生完成实验、课外作业、大作业的情况，反映出学生学真正学会了编程，全面提高学生“码力”，解决了计算机公共基础课“学了程序设计却不会写程序”的问题，为工科专业后续课程打下了编程基础。

3.课程教改效果达成情况

1) **针对众口难调的难点**，P-PMCO 的设计有机融合了各方面的资源，做到充分引导，各取所需，有机融合。

2) **针对平台分散的痛点**，P-PMCO 实现了自研平台嵌入雨课堂，融合一平台，混而不杂，一键可达。

3) **针对高阶育人目标**，P-PMCO 始终贯穿思维引领，实践育人，价值塑造贯，形成了以学生发展为中心，全学期围绕大作业的递进式设计；每个知识点精巧设计层次递进式混合教学；结合全学期不同阶段的需求，实现了混合式教学全流程多角度的设计。

4) **积累形成可移植可复制的翻转课堂方案**，全国范围推广。雨课堂课件、混合式教学教师手册，雨课堂练习题，均可灵活组合，随教材在全国多所院校推广。

以下为 2020 年-2021 年春周三课堂班的数据展示，可以看到资源全部集成于雨课堂，学生参与度高，课堂互动积极，在课后的问卷调查中显示 85%的同学表示收获蛮多，对课程的各个教学环节都给予了认可，并普遍反映大作业的设计让他们收获最大。

雨课堂课件



作业



视频



教学数据统计

课堂

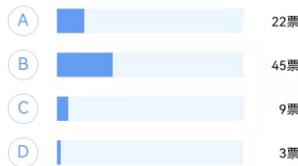


课堂直播



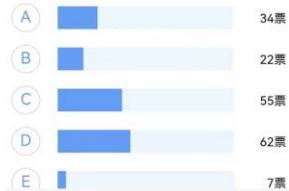
你觉得课程收获如何?

- A 收获非常大
- B 收获蛮多
- C 收获还好
- D 收获不太多
- E 不好说



你觉得哪个环节相比而言更有收获? (请选择不超过3项)

- A 上课听讲
- B 看MOOC
- C 做在线编程OJ
- D 大作业
- E 课前预习
- F 不好说



课程推广示范效应:

1. 本课程的混合式教学经验在全国几十所院校进行分享交流。
2. 本课程作为首批新时代高效教师融合时教学公益进修项目入选课程，来自全国的其他高校的 59 名同课程教师参与全学期的观摩和学习，最终 22 人参加期末考试完成了整个学期的进修流程。其创新式的混合式教学模式受到了同行教师的认可。
3. 雨课堂课件、混合式教学教师手册，雨课堂练习题，均可灵活组合，随教材在全国多所院校推广。

### 三、一次混合式教学设计方案 (与提交的说课视频相对应的一次混合式教学设计方案)

1.学习目标	<p>(本课次学生能够掌握和展现的具体知识、技能和能力等，对课程目标的贡献等情况)</p> <p>课程教学内容：三种基本排序算法和文件读写</p> <p>教学目标：</p> <ol style="list-style-type: none"> <li>1. 知识层面： <ol style="list-style-type: none"> <li>a) 掌握插入排序，选择排序和冒泡排序三种排序算法和具体代码实现；</li> <li>b) 掌握计算机进行文件输入输出流的逻辑和实现；</li> </ol> </li> <li>2. 能力层面： <ol style="list-style-type: none"> <li>a) 理解排序算法的计算思维，具备初步的分析能力；</li> <li>b) 理解软件开发中以文件输出作为程序调试的重要方式；</li> </ol> </li> <li>3. 价值层面： <ol style="list-style-type: none"> <li>a) 初步建立工程性思维，培养严谨务实的做事风格；</li> <li>b) 加强团队合作，锻炼其团队合作意识和技巧；</li> <li>c) 引入竞争机制，加强引导，建立良好的竞争意识和奉献精神。</li> </ol> </li> </ol>
2.内容与资源	<p>(本课次内容与资源的选取、制作、使用情况)</p> <ul style="list-style-type: none"> <li>● MOOC 视频资源：第十章泛型程序设计——排序以及第十一章流类库与输入/输出，总共包括 14 个视频片段和 5 次穿插习题。</li> <li>● 课前发布课前知识点巩固，包括 cin/cout 标准输入输出流操作符的功能，两个数字进行交换的函数写法，复习函数传参和引用的区别。</li> <li>● 文件输入输出基本代码以供学生尝试。</li> <li>● 在线编程 OJ 作业</li> </ul>
3.过程与方法	<p>(本课次混合教学的实施过程与方法)</p> <ol style="list-style-type: none"> <li>3.1 教学方法改革： <ol style="list-style-type: none"> <li>1. <b>突破传统的代码算法讲解无法落地的局限：</b>从算法思路出发，利用翻转课堂从思维方式到课堂做题演练再到代码实践，突破学生思维瓶颈，扎实落实课堂收获。</li> <li>2. <b>以能力培养为中心的混合式教学模式：</b>课前精巧设计，突破以知识点为关键字的引领方式，从知乎上的一片文章出发，引出大作业测试的需求，推出相关视频，初步代码和相关练习，课堂上以思维方式讲解为主线，层层递进，并结合雨课堂的日志和功能来分析知识点的具体实现，在课堂上设计结对编程，锻炼学生软件代码迭代能力的培养，实现了多方面多角度的混合。</li> <li>3. <b>以学生发展为中心的价值塑造过程：</b>以文件输入输出为切入点落实学生对于软件开发流程的设计需求理解，理解作为程序设计者的思维方式和软件服务用户的设计意识；在课堂上进行结对编程，设计相应任务，鼓励最先完成的团队，形成学生的团队意识，提高协作能力。</li> </ol> </li> <li>3.2 教学实施过程： <ol style="list-style-type: none"> <li>1. 课前引导</li> </ol> <p>从知乎上的一片文章出发，引出大作业测试的时候会遇到的困惑，从而引出排序和文件</p> </li> </ol>

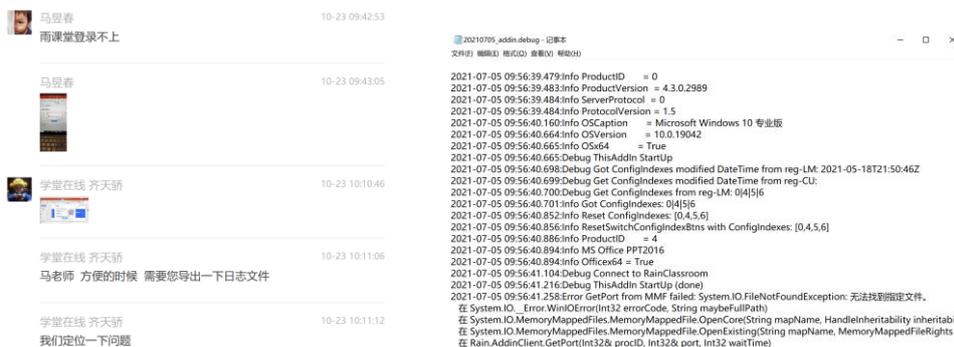
输入输出的需求，并针对基础语法给出相应测试和基础代码，让学生小试牛刀。



- 测试时的困惑
  - 测试用例怎么写?
  - 游戏怎么测试?
  - 我刚才测了什么?
  - 还要写测试报告?
  - .....

## 2. 课堂实施递进式教学

- 针对测试时的困惑引入日志文件读写的需求和设计理念设计，引入文件输入输出流的语法概念；以雨课堂日志为例，通过主观题研讨，大作业设计需要什么样的日志文件？



- 基于大作业项目开发提出排序问题的基本思维：
  - a) 插入排序
  - b) 选择排序
  - c) 交换排序(冒泡排序)



- 以具体形象思维出发，从三支笔的排序开始分别介绍不同方式下的排序思路，再以动图的形式利用雨课堂的填空题加强学生的思路。

- 从思维再进一步拓展到具体的代码实现。结合动图落实代码实现中的数据设计和具体语法知识。

C++ 程序设计 例12\_8 直接插入排序函数模板

```

template <class T>
void insertionSort(T a[], int n) {
    for (int i = 1; i < n; i++) {
        int j = i;
        T temp = a[i];
        while (j > 0 && temp < a[j - 1]) {
            a[j] = a[j - 1];
            j--;
        }
        a[j] = temp;
    }
}

```

- 不同算法的汇总分析

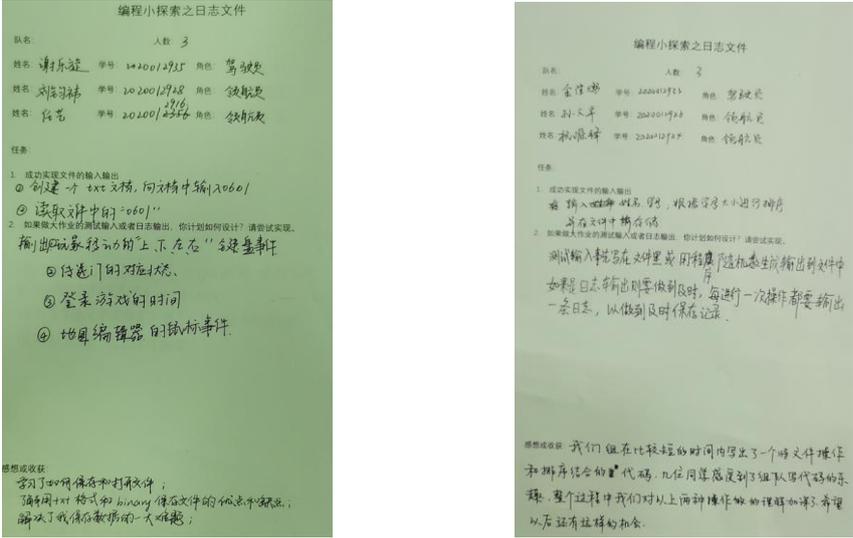
类别	排序方法	时间复杂度			空间复杂度	稳定性
		平均情况	最好情况	最坏情况	辅助存储	
插入排序	直接插入	$O(n^2)$	$O(n)$	$O(n^2)$	$O(1)$	稳定
	shell排序	$O(n^{1.3})$	$O(n)$	$O(n^2)$	$O(1)$	不稳定
选择排序	直接选择	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$	不稳定
	堆排序	$O(n \log_2 n)$	$O(n \log_2 n)$	$O(n \log_2 n)$	$O(1)$	不稳定
交换排序	冒泡排序	$O(n^2)$	$O(n)$	$O(n^2)$	$O(1)$	稳定
	快速排序	$O(n \log_2 n)$	$O(n \log_2 n)$	$O(n^2)$	$O(n \log_2 n)$	不稳定
	归并排序	$O(n \log_2 n)$	$O(n \log_2 n)$	$O(n \log_2 n)$	$O(1)$	稳定
	基数排序	$O(d(r+n))$	$O(d(n+r))$	$O(d(r+n))$	$O(rd+n)$	稳定

注: 基数排序的复杂度中,  $r$ 代表关键字的基数,  $d$ 代表长度,  $n$ 代表关键字的个数

3. 利用一节课的时间安排学生结对编程, 安排相应的任务清单。对完成任务的同学颁发小红旗以资鼓励。

4. 课程思政嵌入点:

- 思维引领: 利用“三支笔”的场景设计, 引入不同的排序算法, 引导学生思考对学科思维方式的体会, 理解从现实问题到抽象算法再到复杂问题求解实现的思维历程;
- 严谨扎实: 通过雨课堂日志的分析, 体会程序员应有的担当和意识, 要有全局观和风险防范意识;
- 为谁学习为谁忙: 通过嵌入大作业需求, 让学生体会到程序设计的落地需求, 并通过结对编程形成团队合作情境, 通过引入竞争意识鼓励完成任务发放国旗, 让学生自然联想到“要又红又专地编好程序”。

<p>4.评价与反馈</p>	<p>(本本次学习评价与反馈方式)</p> <p>本次学习课堂上的互动题目由雨课堂自动完成，课后的评价主要体现在大作业的完成度上。课堂翻转过程中采用了任务单的形式，课后回收任务单，可以得到一手的主观反馈和学生实施的进度情况。因此通过雨课堂实时数据，弹幕信息，以及视频观看情况，习题解答情况相互综合，教师可以及时，准确而且全面的掌握学生的学习情况，实现精细化教学和对后续教学改良提供有效数据支持。</p>  <p>The image shows two screenshots of student task sheets for a programming assignment titled '编程小探索之日志文件' (Programming Exploration - Log File). The sheets list student names, IDs, and roles, and contain handwritten task instructions and solutions. The first sheet lists tasks like creating an output file, reading '0601' from a file, and logging events like login and map editing. The second sheet shows a student's handwritten solution for these tasks, including code snippets and a reflection on the learning process.</p>
<p>5.教学效果达成情况</p>	<p>(本本次教学效果与特色)</p> <ol style="list-style-type: none"> <li>1. 优质而丰富的在线资源建设实现了大班课堂里的因材施教，传统的教学讲解以上内容至少需要 3 个学时，而通过线上线下相结合的方式使在线资源“活起来、用起来”，充分引导学生的学习需求，使得我们用一节理论讲解一节课翻转课堂结对编程实现了高效率的翻转教学，学生普遍反映收获大，效率高。</li> <li>2. 精巧的教学设计具有很强的层次递进性，落实了布鲁姆的教学目标分类从记忆，理解到应用，甚至创新的完整流程，实现了高阶的教学目标，并在教学设计中留有活口。从三支笔到三个动图，再从动图到代码实现，从实现中对比三种代码的优劣，再开放式的引入多种常用算法的广阔空间，为未来后续课程《数据结构与算法》的学习进行铺垫，为专业发展奠定基石。</li> <li>3. 教学贴近生活，形象思维和抽象思维相互穿插，让学生感慨原来日志文件要做到事无巨细，什么都记；从信手拈来的三支笔到高深复杂的算法比较，让学生体会抽象之美；翻转课堂里的角色扮演，竞争机制，让学生体会到团队合作，责任和未来担当。</li> </ol>