**《计算思维与实践》课程教学大纲**

一、课程基本信息

课程代码：16211601/16211602

课程名称：计算思维与实践

英文名称：Computational Thinking and Practice

课程类别：通识选修课

学 时：16/32

学 分：1/2

适用对象: 非计算机类专业本科学生

考核方式：考查

先修课程：无

二、课程简介

**中文简介**

计算思维是运用计算学科的基础概念进行问题求解、系统设计、以及人类行为理解的思维活动，计算思维是的培养是信息时代对人的基本要求，因此在大学本科阶段开设这门课程是十分重要和必要的。《计算思维与实践》课程是一门计算机类通识型课程，也是高等学校计算机基础教学公共课程之一 。该课程从培养学生科学认知能力出发，让学生理解和建立“信息、计算、智能”这三大核心科学概念；围绕计算思维的精髓培养学生掌握以“合理抽象、高效实现”为特征的构造性过程的能力；让学生了解学科发展，展示计算之美。初步达到传承计算文化、弘扬计算之美、培养计算思维的目的。

**英文简介**

This course of computational thinking and practice is about computational disciplines of basic concepts of problem solving, system design, and human behavior understanding of thinking activity, computational thinking is the training is basic requirements of the information age to, therefore in the undergraduate stage opened this course is very important and necessary.

The course is a normal computer course, but also one of the basic public course of Computer Science in Colleges and Universities. The purpose of the course is for the cultivation of students' cognitive ability and make students understand and establish the core scientific concepts：information, computing, intelligent and so on. The course is around the calculated the essence of thinking to train the students to master the reasonable abstraction, efficient implementation feature construction process; It make students understand discipline development, showing computational beauty. The purpose of the course is inheriting the culture of calculating, carrying forward the calculation of the United States, and cultivating the computational thinking.

三、课程性质与教学目的

本课程除了需要介绍计算思维理论，还需要处理好理论和实践之间的关系。在理论上有一定的深度和难度，如何利用实践教学的环节帮助学生掌握理解基本概念和基本原理是教学的重要环节。只有通过精心设计的实验才能使学生掌握信息、计算、智能基本概念，以及掌握求解问题的基本思路和能力。同时，让学生在编写程序的过程中体会到：**程序中的每一条语句都要遵守语法规则，每一条语句都有它存在的意义时，引导学生思考自己作为青年大学生，要严格遵守疫情期间国家和学校的指示，要勇敢回应时代的挑战，要勇做历史使命和责任的传承人。**

本课程的教学目的是通过作为轻语法程序设计典型代表的Python语言为载体，重点训练学生理解并实践计算思维，掌握解决计算问题的能力。主要培养过程包括如下四个阶段：

第一阶段：计算机的组成与工作原理，包括计算机的起源、计算机的特征、计算机的工作过程、计算机程序、程序设计语言、计算机程序中数的表示等，加深学生对计算机的理解

第二阶段：计算思维的思想与方法，包括计算思维概念理解，基本的程序设计方法、Python语言基本语法等，训练学生通过程序设计理解计算问题求解。

这部分涉及的计算思维概念包括：抽象、自动化、程序、算法等。这些概念讲解融入Python语言语法和程序设计基本方法中，包括用Python语言实现分支结构和循环结构，函数调用以及数组与字典的实现方法 。

第三阶段：问题的抽象与算法设计。进一步丰富计算思维的内涵和外延。训练学生从问题求解的高度理解计算思维。

这部分涉及的计算思维方法包括：穷举法、递归法、分治法、回溯法和贪心法等。

第四阶段：复杂计算问题的求解过程和算法比较。通过几个类型实例运用计算思维，结合算法分析和比较等训练学生运用计算思维求解复杂计算问题。

四、教学内容及要求

第一章 计算机的组成与工作原理

1. 目的与要求
2. 认知目标：使学生认识、了解计算机系统的组成及简单工作原理。
3. 技能目标：培养学生独立思考和寻求求解方法的能力。
4. 能力目标：培养和激发学生学习程序设计的兴趣，促进其个性发展。
5. 教学内容
   * 1. 计算机的起源
6. 主要内容
7. 计算机发展史
8. 中国计算机的发展史

中国计算机事业的起步比美国晚了13年，但是经过老一辈科学家的艰苦努力，中国与美国的差距在不断得缩小。

1. 计算机应用领域
2. 基本概念和知识点

电子管、晶体管、中小规模集成电路、大规模集成电路、超大规模集成电路。

1. 问题与应用（能力要求）

问题：

* 生活中，计算机无处不在，计算机能帮我们做什么？为什么计算机能做这么多事情？
* 请预测一下，对于你们各自的专业，计算机哪些领域是你所学专业未来发展的必然趋势？（科学计算（早期）、办公自动化、数据处理与分析（信息管理）、生产自动化（过程控制）、辅助设计、多媒体应用、人工智能、计算机网络等）
* **讲到办公自动化软件，引入求伯君开发国产文字编辑软件WPS从而打破垄断的艰辛历程。（增强对国产软件的自豪感）**

要求学生完成本节学习后，对计算机的发展过程有个初步的了解。

* + 1. 计算机的特征

1. 主要内容
2. 计算机工作原理
3. 计算机组成
4. 计算机的工作过程
5. 基本概念和知识点

运算器、控制器、存储器、输入输出设备、存储程序、执行指令、数据总线。

1. 问题与应用（能力要求）

问题：计算机各组件主要的功能分别是什么？

要求学生完成本节学习后，对计算机组成和特征有个粗略的了解。

* + 1. 计算机程序设计

1. 主要内容
2. 什么是计算机程序
3. 程序设计语言
4. 各种程序设计语言的对比
5. Python语言的特点和优势
6. Python计算生态
7. Python的简单语法和基本结构
8. 基本概念和知识点

程序设计语言的语法和语义、程序设计语言的特点、两类高级语言的差异、设计程序的过程、脚本语言、Python语言初识。

1. 问题与应用（能力要求）

问题：为什么要学习程序设计？

要求学生完成本节学习后，对程序和程序设计语言有个初步的认识并熟悉其操作界面。

**由于Python语言通过开源模式，构建了庞大的生态系统，从数据分析、大数据处理、机器学习、网络编程等领域均有广泛的应用，其中不乏国内开发的优秀工具库，如中文分词第三方库Jieba和SnowNLP（明确学习Python编程的意义，增强学生的民族自豪感和爱国热情。）**

1. 思考与实践

让学生课后通过浏览一些网页来了解计算思维的基础知识；并通过实践掌握简单使用Python语言设计程序来解决问题的方法。

1. 教学方法与手段

本章教学主要借助多媒体课件采用课堂讲授的方式进行。

* 1. 计算机思维的思想与方法

1. 目的与要求
2. 掌握问题抽象的方法。
3. 理解计算机算法以及复杂度的分析。
4. 熟练掌握利用Python语言实现分支与选择结构、循环结构、函数调用以及数组与字典的创建和应用。
5. 初步利用Python语言求解简单计算问题的方法。
6. 教学内容
   1. 什么是计算思维
7. 主要内容
8. 计算思维的操作模式
9. 计算思维的作用
10. 运用计算机求解问题的基本过程和优势
11. 计算思维的经典案例
12. 生活中的计算思维
13. 计算思维与各学科的关系
14. 基本概念和知识点

自动化、预取与缓存、符号、问题抽象与建模、近似解，随机化过程、启发式方法、递归思维、并行处理、回推过程；

1. 问题与应用（能力要求）

问题：举例说明生活中存在哪些计算思维？例如，红绿灯设计、超市付账时收银台的选择。

要求学生完成本节学习后，掌握简单计算问题抽象与建模的过程，熟悉用计算思维方式解决简单计算问题，并寻求计算思维与学生自身专业的关系。

* 1. 问题求解与计算机算法

1. 主要内容
2. 什么是计算机算法
3. 算法的伪代码藐视
4. 程序设计过程
5. 利用Python语言验证计算思维
6. Python语言的基本语法和语义
7. Python语言的数据类型和表达式
8. Python程序基本调试过程
9. 程序举例
10. 良好的编程风格
11. 程序流程图
12. 基本概念和知识点

数学思维与计算思维、排序、查找、计算速度、伪代码、需求分析、算法设计、代码实现、编译调试、数据结构、运算符和表达式、赋值语句、输入输出、流程控制结构、函数、数组、关键字、变量和常量、注释、流程图。

1. 问题与应用（能力要求）

问题：程序调试的方法有哪些？

要求学生完成本节学习后，熟练掌握Python的语法和语义，能利用Python实现简单计算问题的求解过程。

**程序顺序结构的特点一战到底，没有选择，没有重复，引导学生珍惜时间， 自觉学习， 磨炼学生坚持不懈、 不忘初心、砥砺前行的思想品质。**

* 1. 分支与选择结构

1. 主要内容
2. 单分支的实现
3. 双分支的实现
4. 多分支的实现

**学习选择结构时， 引导学生去联想在中国疫情最严峻的时刻，那些曾经在父辈护佑下的90后00后青年医生护士们，他们在面临居家与抗疫第一线的选择时，是如何选择的？他们选择了后者，选择了在挥汗如雨中脱胎换骨，在逆境挑战中顶天而立，他们一边恐慌，一边在勇敢中破茧成蝶，淬炼成钢。**

1. 基本概念和知识点

Python语言中if语句、分支条件、分支程序的复杂度分析；

1. 问题与应用（能力要求）

问题：如何利用分支与选择结构？

要求学生完成本节学习后，熟练掌握分支与选择结构的实现。

* 1. 循环结构

1. 主要内容
   * 1. 循环结构程序的编写方法
     2. for循环机制
2. 基本概念和知识点

计数器、循环次数、循环控制条件、浮点数、循环结构对算法复杂度的影响

1. 问题与应用（能力要求）

问题：如何利用普通表格归整表单？

要求学生完成本节学习后，熟练掌握网页基本元素（包括多媒体对象、表格及表单等）的插入及设置。

* 1. 函数

1.主要内容

* + 1. 程序模块化
    2. 什么是函数
    3. 函数调用
    4. 关于函数参数
    5. 函数应用
    6. 函数的递归调用

2.基本概念和知识点

程序模块、模块化、参数、值传递、递归

3.问题与应用（能力要求）

问题：如何利用函数运行的结果？

要求学生完成本节学习后，熟练掌握自定义函数的方法和函数调用的过程。

* 1. 数组和字典

1.主要内容

* + 1. 一维数组
    2. 字符串数组
    3. 二维数组
    4. 字段
    5. 数组应用

2.基本概念和知识点

一维数组、数组元素、数组下标、字符串、二维数组、字典

3.问题与应用（能力要求）

问题：如何利用数组和字典批量处理数据？

要求学生完成本节学习后，熟练掌握数组应用方法。

* 1. 问题的抽象与算法设计

1. 目的与要求
2. 了解算法复杂度的含义及分析方法；
3. 掌握经典的集中算法设计方法；
4. 教学内容

第一节 算法复杂度

1. 主要内容
2. 算法的评价
3. 时间复杂度
4. 空间复杂度
5. 基本概念和知识点

算法评价标准、计算机资源、容错能力、问题规模、算法效率。

1. 问题与应用（能力要求）

问题：如何设计出最优的算法？

要求学生完成本节学习后，能够对算法复杂度有了清晰的了解。

第二节 穷举法

1. 主要内容
2. 穷举法的特点
3. 穷举法的应用
4. 基本概念和知识点

解的空间范围

1. 问题与应用（能力要求）

问题：如何确定解的空间范围？

要求学生完成本节学习后，能够利用Python语言实现穷举算法。

第三节 递归法

* 1. 主要内容

1. 递归法的特点
2. 递推法的应用
3. 公式法的应用
   1. 基本概念和知识点

问题规模分解、直接调用、间接调用。

* 1. 问题与应用（能力要求）

问题：如何确定递归的结束条件？

要求学生完成本节学习后，能够利用Python语言实现递归算法。

第四节 分治法

1.主要内容

1. 分治法的特点
2. 分治法的应用

2.基本概念和知识点

分而治之、二分查找、归并排序，Python语言中列表、树状结构、树的层数

3.问题与应用（能力要求）

问题：如何构造树状的结构？

要求学生完成本节学习后，能够利用Python语言实现分治算法。

第五节 回溯法与贪心法

1.主要内容

1. 回溯法的特点
2. 回溯法的应用
3. 贪心法的特点
4. 贪心法的应用

2.基本概念和知识点

深度优先搜索、广度优先搜索、解的空间结构、冲突、全局最优解、局部最优解、目标函数、约束条件

3.问题与应用（能力要求）

问题：如何将新问题归结于老问题的方法来求解？

要求学生完成本节学习后，能够利用Python语言实现回溯法和贪心法。

1. 思考与实践

思考题：学习经典问题的求解方法有何好处？

让学生能利用Python实现各种经典算法。

1. 教学方法与手段

本章教学主要借助多媒体课件采用课堂讲授和案例分析的方式进行。

* 1. 最优化问题

1. 目的与要求
2. 了解最优化问题求解的复杂度
3. 熟练掌握Python语言实现几个经典最优化问题的算法
4. 教学内容
5. 主要内容
6. 最短路径问题
7. 游商问题
8. 装箱问题
9. 背包问题
10. 生活中出现的最优化问题
11. 各种方法求解最优化问题的比较
12. 基本概念和知识点

NP完全问题、动态规划、重叠子问题、Python语言中的记录字段

1. 问题与应用（能力要求）

问题：简述利用Python语言的求解计算问题的过程。

要求学生完成本节学习后，能够对本地网站建设流程有个更为清晰的了解。

1. 思考与实践

思考题：如何判断算法是否最优？

让学生通过实践来掌握Python语言实现最优化问题的算法。

1. 教学方法与手段

本章教学主要借助多媒体课件采用课堂讲授的方式进行。

五、各教学环节学时分配

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **教学环节**  **教学时数**  **课程内容** | **讲 课** | **习 题课** | **讨论课** | **实验** | **其他教学环节** | **小计** |
| 第一章 计算机的组成与工作原理 | **4** |  |  | **2** |  | **6** |
| 第二章 计算机思维的思想与方法★ | **4** |  |  | **4** |  | **8** |
| 第三章 问题的抽象与算法设计★ | **4** |  |  | **4** |  | **8** |
| 第四章 最优化问题 | **4** |  |  | **4** |  | **8** |
| 综合测试 |  |  |  |  | **2** | **2** |
| 合计 | **16** |  |  | **14** | **2** | **32** |

注：若是16学时，则只讲表中带★的部分

六、课程考核

（一）考核方式：课程设计

（二）成绩构成

平时成绩占比：40% 期末考试占比：60%

（三）成绩考核标准

平时成绩参照考勤、课堂表现、作业和实验报告四方面的情况给出。

期末考试：要求学生自主完成一个选题，并撰写课程设计报告，

七、推荐教材和教学参考资源

1. 推荐教材：

《Python程序设计与实践——用计算思维解决问题》，清华大学出版社出版，李莹主编，ISBN：978-7-302-47389-3

《计算思维导论——一种跨学科的方法》，清华大学出版社出版，李暾编著，ISBN： 978-7-302-44225-7

1. 参考书目：
2. 《Python\_3.4\_入门指南(官方中文版)》《Python 学习手册》（中文版）机械工业出版社 ISBN：978-7-111-32653-3 [美] 鲁特兹著；李军等译；
3. 《计算思维与算法设计》，人民邮电出版社出版， 麻新旗、王春红编；
4. 《计算思维应用实例》，清华大学出版社 赵宏编；
5. 《程序设计思想与方法——问题求解中的计算思维》， 高等教育出版社。

大纲修订人：黄元南 修订日期：2022年1月

大纲审定人： 审定日期：