

《典型运筹学问题与模型》教学设计

课程名称	第二章 背包问题及经典求解算法	学时	3
学情分析	<p>典型运筹学是一个深入研究优化决策过程的经典领域。自古以来，人们对如何优化资源配置和提升效率充满兴趣。在现代社会，运筹学应用广泛，从物流配送到生产调度、从金融投资到交通流量管理，这些应用的成功都离不开运筹学的支持。运筹学通过数学模型和算法，帮助我们解决各种复杂的决策问题，例如最优路径选择、资源分配和系统优化等。在实际应用中，它不仅依赖于领域知识和问题的精确建模，还涉及高效的算法设计和解决策略。通过系统地解决这些问题，运筹学为各种行业提供了有力的决策支持。</p> <p>授课对象为大数据实验班，该班级学生普遍具备扎实的数学和运筹学基础，对运筹学课程有较高的认知水平。学生的学习风气良好，思维活跃，具有较强的好奇心和进取心，这为深入探讨运筹学的理论和应用打下了坚实的基础。由于他们已经具备了良好的基础知识，这使得他们能够更快地理解和掌握课程中的复杂概念和高级算法，能够积极参与讨论和解决实际问题。因此，在授课过程中，可以更深入地探讨运筹学的高级话题，结合实际案例和大数据应用，进一步提升学生的实际操作能力和理论水平，以满足他们对未来工作的高要求。</p>		
教学目标	<p>1. 掌握背包问题的特征及求解算法： 学生能够理解背包问题的基本概念，熟练运用动态规划和贪心算法等多种方法解决不同类型的问题，分析算法的优缺点。</p> <p>2. 了解背包问题的变形： 学生识别背包问题的多种变形，并结合实际应用，提高分析解决复杂问题的能力。</p> <p>3. 爱国情怀与社会责任感： 通过案例引入，增强学生对科技创新的认同，激励他们关注社会发展，培养创新精神与责任意识。</p>		
教学思想	<p>充分调动学生学习主动性和积极性</p> <p>用讨论交流的方式引导学生梳理知识</p> <p>将立德树人、德智兼修的思想融入课程教学中</p> <p>提高学生专业能力的同时帮助学生树立正确的人生观、价值观。</p>		
课程资源	<p>《组合最优化：理论与算法》 科学出版社 第 17 章</p> <p>《数学规划及其应用》范玉妹，徐尔，赵金玲，胡毅庆 机械工业出版社 第 7 章（7.5.4）</p> <p>MIT OpenCourseWare - Introduction to Algorithms (6.006) 第 2 章</p> <p>课件 源代码</p>		
教学内容	<p>1. 思政案例引入</p> <p>2. 基本背包问题特征及数学模型</p> <p>3. 背包问题的变形（特征、模型描述）</p> <p>4. 基本背包问题的求解算法</p>		
教学重点与难点	<p>重点：基本背包问题数学模型与求解算法</p> <p>难点：0-1 变量的理解和使用</p>		
教学方法与工具	<p>教学方法：本科主要采用任务驱动教学，过程中辅以启发提问、自主学习方法</p> <p>工具：多媒体、板书</p>		
教 学 安 排			
教学环节	教师行为	预设学生行为	设计意图
案例导入（5 分钟）	分享 人员配餐过程中的送餐路线、后备箱装载 等案例，展现典型运筹学在现实生活中的应用，进而引出什么是背包问题	回答什么是背包问题	引入典型运筹学在生活场景中的应用，培养学生社会责任感 ，激发学习意识；了解学生对背包问题的认知程度
知识讲授（60 分钟）	1. 背包问题的特点、数学模型，重点讲解 0-1 变量，如何使用 0-1 变量 2. 介绍背包问题的变形，如多选择背包问题、多	关注教师授课内容	使学生 掌握背包问题的特点及数学模型 ，并熟练运用 0-1 变量

	<p>约</p> <p>束背包问题的数学模型</p> <p>3. 背包问题的应用</p> 		
课堂提问（5 分钟）	提问：小结刚才的授课内容	回答问题	加深对知识点的记忆
<p>基本背包问题的求解方法：分支定界法。先让学生自主学习，独立思考，之后再进行讲解。（20 分钟）</p>	<p>分支定界法的组成</p> <p>分枝：在 B 的最优解处分别增加两个整数约束，将可行域 B 分枝成两个子区域 B1 和 B2</p> <p>剪枝：剪枝 B2，如何剪？什么规则</p> <p>定界：在什么结点定，如何定</p>	<p>1. 理解基本概念：学生应了解分支定界的基本理论，包括如何通过树形结构和界限函数来解决优化问题。</p> <p>2. 掌握算法流程：学生需熟悉分支定界算法的具体步骤，包括节点扩展、界限计算和剪枝策略，以便高效地求解问题。</p>	<p>1. 确保学生理解分支定界算法的基本原理和流程，包括如何设定界限和选择分支策略。</p> <p>2. 通过具体实例和案例，帮助学生掌握算法在实际问题中的应用，增强其解决复杂优化问题的能力。</p> <p>3. 培养学生的编程和调试能力，使他们能够独立实现和优化分支定界算法。</p>
<p>基本背包问题的求解方法：贪婪算法。先让学生自主学习，独立思考，之后再进行讲解。（25 分钟）</p>	<p>背包问题的贪婪启发式</p> <p>对背包问题可以构造下面的贪婪算法：</p> <p>Step1: 对物品以 $\frac{C_i}{a_i}$ 从大到小排列，不妨把排列记成 $\{1, 2, \dots, n\}$，$k=1$；</p> <p>Step2: 若 $\sum_{i=1}^{k-1} d_i x_i + d_k \leq b$，则 $x_k=1$，否则，$x_k=0$；$k=k+1$</p> <p>当 $k=n$ 时，停止；否则，重复 Step2</p>	<p>1. 掌握贪婪算法策略：学生应理解贪婪算法的核心思想，即在每一步选择局部最优解，来尝试得到整体的最优解。对于背包问题，常见的贪婪策略包括按物品的单位价值（价值/重量）排序并选择高单位价值的物品。</p> <p>2. 实现算法：学生会编写代码实现贪婪算法解决背包问题，并对不同的输入实例进行测试，以验证算法的正确性和效果。</p>	<p>1. 帮助学生掌握贪婪算法的基本理念，即通过每一步的局部最优选择来构建全局最优解的策略。</p> <p>2. 培养学生对贪婪算法性能的分析能力，包括时间复杂度和空间复杂度的理解，以及贪婪算法在背包问题中的适用范围和局限性。</p>
<p>基本背包问题的求解方法：邻域搜索算法、遗传算法、动态规划。针对每一种算法，先让学生自主学习，独立思考，之后再进行讲解。（45 分钟）</p>	<p>简单介绍邻域搜索算法、动态规划。</p> <p>主要介绍遗传算法，其求解方法分类为：二进制表达法 (Binary representation)；顺序表达法 (Order representation)；变长表达法 (Variable-length representation)。</p> <p>讲解遗传算法的核心概念，包括选择、交叉、变</p>	<p>1. 学习遗传算法基本概念：理解选择、交叉、变异和适应度函数等核心概念。</p> <p>2. 实现编码和解码方法：理解如何将解转换回实际问题中的解。</p>	<p>1. 理解遗传算法基础：使学生掌握遗传算法的核心原理，为解决背包问题奠定理论基础。</p> <p>2. 应用算法于实际问题：引导学生将遗传算法应用于背包问题，提升其解决实际优化问题的能力。</p>

	<p>异和适应度评估，并说明如何将这些概念应用于背包问题。</p> <p>详细介绍遗传算法的各个步骤，包括初始种群的生成、适应度函数的设计、选择操作、交叉操作、变异操作和终止条件。解释每一步如何影响算法的性能和结果。</p>	<p>3. 定义适应度函数: 创建适应度函数来评估每个个体的背包总价值，并处理约束条件（如重量限制）。</p> <p>4. 设置和调整算法参数: 选择合适的种群大小、交叉率和变异率，并进行实验以优化这些参数。</p>	<p>3. 培养编程技能: 通过实现算法，增强学生的编程能力和调试技能。</p>
内容总结（5 分钟）	请学生对上述内容进行总结	总结教师讲授内容	了解并加强学生对知识的掌握
教学评价	<p>形成性评价：通过课堂中的小测验、即时反馈、讨论参与等方式，实时评估学生的学习进度。这种评价方式注重过程，帮助教师在教学过程中发现学生的困难与不足，并及时进行调整。</p> <p>总结性评价：在单元或课程结束时，通过考试、论文、项目报告等方式评估学生的最终学习成果。注重学生对知识的整体把握和实际应用能力。</p> <p>过程性评价：通过观察学生在课堂中的参与度、作业完成情况、项目进展等过程性表现，综合评估学生的学习状态和进步。</p> <p>结果性评价：结果性评价重点关注学生通过学习最终达成的成果，如期末考试成绩、项目完成情况等。这类评价能够反映学生的总体学习效果和 能力发展。</p> <p>课堂氛围活跃，学生积极回答问题，对知识点总结到位，知识掌握水平较好</p>		
预习任务	预习： 什么是装箱问题？		
课后作业	要求完全掌握基本的背包问题数学模型，对求解算法上机操作		
板书设计	<div><div>背包问题</div><div><div>案例导入</div><div>展现典型运筹学在现实生活中的应用，进而引出什么是背包问题</div></div><div><div>背包问题的特点、数学模型</div><div>讲解0-1变量，如何使用0-1变量</div></div><div><div>背包问题的变形</div><div>多选择背包问题(multiple-choice knapsack problem)</div><div>多约束背包问题(multi constrained knapsack problem)</div><div>有界背包问题(bounded knapsack problem)</div><div>无界背包问题(unbounded knapsack problem)</div></div><div><div>基本背包问题的求解方法</div><div>分枝定界法</div><div>贪婪算法</div><div>邻域搜索算法</div><div>动态规划</div><div>遗传算法</div></div><div>对背包问题可以构造下面的贪婪算法： Step1: 对物品以 $\frac{c_i}{a_i}$ 从大到小排列，不妨把排列记成 $\{1, 2, \dots, n\}$，$k=1$； Step2: 若 $\sum_{i=1}^{k-1} d_i x_i + d_k \leq b$，则 $x_k = 1$，否则，$x_k = 0$；$k=k+1$ 当 $k=n$ 时，停止；否则，重复 Step2</div><div>二进制表达法(Binary representation) 顺序表达法(Orderepresentation) 变长表达法(Variable-length representation)</div><div>包括初始种群的生成、适应度函数的设计、选择操作、交叉操作、变异操作和终止条件。</div></div>		