课程模块	研讨主题	相关知识点	研讨过程	备注
	1 从莱布尼兹的梦想到图灵的通用计算机（理论，历史，思想） 探讨数理逻辑的发展过程中相关思想方法，理解数理逻辑在计算机科学中的基础作用. 主要参考资料：逻辑的引擎（图书）.	普遍语言，符号逻辑，判定问题，通用计算机模型，思想，方法.	1 提出问题 2 学生尝试解决问题 3 小组组织若干次讨论并提交结果（内容可以主要是问题） 4 师生交流 5 小组写报告或小论文（长短不限，内容可以是综述，实验，示例，总结等） 6 提交，反馈	
数理逻辑	2 对偶定理（理论） 主要探讨定理的证明，可扩充了解数学中对偶定理. 参考： https://baike.baidu.com/item/对偶定理 地大版教材命题逻辑对偶定理一节.	命题逻辑：逻辑联结词，命题公式等值，对偶原理，数学归纳法.		
	3 归结原理（理论与应用） 探讨归结原理（包括正确性证明）及其应用. 扩展：了解逆归结，归纳逻辑程序设计(ILP) 参考：有关教材，或 ILP30 年综述 https://arxiv.org/abs/2008.07912 戴望州 周志华，归纳逻辑程序设计综述 https://crad.ict.ac.cn/CN/10.7544/issn1000-1239.2019.20180759 用 Python 实现命题逻辑归结推理系统 https://blog.csdn.net/Zhangguohao666/article/details/105471307 归结原理- -- 1 条规则 https://blog.csdn.net/Suyebiubiu/article/details/103145475 Propositional Resolution	命题逻辑，谓词逻辑：析取范式，演绎推理，命题符号化，机器证明，机器学习. 可以不深入考虑谓词逻辑中的归结.		

	http://logic.stanford.edu/intrologic/notes/chapter_05.html			
	4 关系演算 E.F.Codd 提出关系数据库模型是当前主流数据库管理系统的数学基础，而关系模型中的关系演算以数理逻辑中的谓词演算为基础。本课题拟探讨谓词演算在关系数据库中的应用。 参考：关系演算 https://blog.csdn.net/hpdlzu80100/article/details/90744715 几篇论文：谓词逻辑在关系数据库中的应用（卢延鑫），元组关系演算的语义研究（卢小兵），用关系谓词推演实现复杂全称量词的结构化查询（钱哨）	数理逻辑，谓词演算，关系数据库		

	5 算法/程序逻辑分析（理论与应用）
	（1）算法/程序等价性验证：现有 2 个关于判断某年份是否为闰年的算法，试判断其是否等价；
	算法 1：
	input: year
	output:“Yes”/ "No”
	if (year=0 (mod 400), or year=0 (mod 4) and (year!=0 (mod 100))
	then return "Yes";
	else return "No".
	注：year=0 (mod 400)表示 year 与 0 模 400 同余，即 year 是 400 的倍数.
	算法 2：
	input: year
	output:“Yes”/ "No”
	if (year=0 (mod 400)) then return "Yes";
		1 命题逻辑：这里的等价定义为有相同的合法输入可得到相同的输出。将算法关键部分（输入条件转换为命题逻辑），证明其在同同真假情况下对应有同样的输出。
		2 谓词逻辑：循环不变式是程序设计理论的重要研究内容，可以用来辅助验证或证明程序的正确性。程序或算法中循环都可以找到一个循环不变式（谓词公式），该循环不变式在初始化（第一代迭代之前）时是为真的，且若每次一次迭代之后也是为真（特别地，在循环终止时也真），从而通过这个循环		

	if (year=0 (mod 100)) the return "No" ;
	if (year=0 (mod 4)) then return "Yes" ;
	else return "No".
	（2）程序/算法中的循环不变式：
	程序 1：计算阶乘
	def factorial(number):
	Fact = 1
	i = 1
	while i<number
	i=i+1
	Fact=Fact*i
	end while
	return Fact
	程序 2：求数组元素最大值
	def max(A):
	answer = A[0]
	for j in range(1,len(A))
	if (A[j]>answer): answer = A[j]
	return answer
	参考：程序正确性证明及循环不变式的寻找方法（王彩芬）
	（3）逻辑演算与程序优化（理论与应用）
	程序优化是高质量软件开发的重要工作，基于命题逻辑等值演算，可以帮助简化程序逻辑，减小程序复杂度，提升程序性能.
		不变式证明循环迭代的正确性。
		3 命题逻辑、谓词逻辑在程序优化、数据库优化中的理论基础作用与应用。		

	参考：教材例 3.3.8			
	6 工程任务计划安排（理论与应用） 某大型建筑公司子公司下有 5 个建筑工程队，现在为完成一项紧急任务，需要安排其中若干个工程队参加。为达成最佳效益和效率，现在根据各个工程队的平时情况（特别是工程队之间的协调合作情况）来安排一种建设任务组队方案。方案具体要求： (1) 如果甲工程队去，则乙、丙工程队至少要去其一； (2) 丙、丁工程队最多能去其一； (3) 戊、乙工程队有且仅有其一参加； (4) 若丁工程队不去，那么甲工程队也不去。 扩展：考虑范式求解的具体实现；了解 SAT 及其应用.	命题逻辑，约束可满足问题（SAT）。从问题构造命题公式，并求其主范式，分析求解。若变量比较多，最好编写程序实现。 可以扩展应用，如基于 SMT 的 AWS 的安全证明。SMT (Satisfiability modulo theories) 在形式化验证、程序语言、软件工程、以及计算机安全、计算机系统等领域得到了广泛应用。		
	7 密码锁电路逻辑表达式求解（理论与应用） 设 P, Q, R, S 为某保密锁的四个按键，当每个按键被单独按下时，锁既不打开也不报警，只有当有 4 个或者 3 个按键分别同时按下时，锁才能被打开；当不符合上述组合状态时，将发出报警信息，试求解此保密锁电路的逻辑表达式，并仅用最少的非门、与非门实现。 参考：数字逻辑组合电路设计. 扩展：考虑求解的具体实现；了解 SAT 及其应用；全加器逻辑设计.	命题逻辑：逻辑抽象、真值表、逻辑函数构造、主范式构造、逻辑公式等值演算。具体根据真值函数（真值表）求解范式，进而求解组合电路的逻辑表达式及其物理实现。		
	8 命题逻辑演算系统的计算机实现 基本要求：实现一个简单的命题逻辑演算系统，具体包括如下功能： (1) 命题公式构造与语法判断：输入仅含否定、析取与合取运算的表达式，判	命题公式，真值表，公式类型，主范式，命题公式推理，推理形式化，计算机编程.		

	断其是否为命题公式，并给出其构造过程（用二叉树）； (2) 计算该命题公式（需要考虑树的遍历问题，并用栈结构进行计算）； (3) 构造公式真值表； (4) 判断公式类型； (5) 构造公式主范式； (6) 应用上述工作，判断给定的推理形式是否正确.			
	9 软件安全与形式化验证 探讨数理逻辑在软件形式化验证中的应用，如，航空航天软件系统的设计与实现，最为重要的就是如何保证其系统核心模块的正确性. 参考：形式化方法概貌（王 戟等，软件学报，2019，30（1））. 芯片开发功能验证的形式化方法（姚广宇等，软件学报，2021，32（6））.	数理逻辑，非经典逻辑，软件/硬件形式化验证		