

04	人工智能	248
05	自主智能系统	250
06	系统工程理论及方法	253
07	系统科学与工程	256
08	检测技术与自动化	258
09	导航与制导系统	260
10	模式识别与机器学习	263
0812	计算机科学与技术一级学科研究生核心课程指南	266
01	高级算法设计与分析	266
02	高级计算机系统结构	268
03	高级分布式系统	271
04	计算机程序理论与模型	274
05	软件系统与工程	277
06	数据科学与工程	280
07	高级计算机网络	283
08	网络与信息安全	286
09	机器学习	289
10	学科前沿与实践	292
0813	建筑学一级学科研究生核心课程指南	295
01	现代建筑理论(与专业学位 0851-4 内容相同)	295
02	建筑评论(与专业学位 0851-05 内容相同)	297
03	建筑历史与理论专题(与专业学位 0851-06 内容相同)	299
04	建筑遗产保护专题(与专业学位 0851-07 内容相同)	300
05	建筑技术科学前沿(与专业学位 0851-08 内容相同)	303
06	数字建筑理论与方法(与专业学位 0851-09 内容相同)	304
07	城市设计理论与方法(与专业学位 0851-10 内容相同)	305
08	建筑策划与使用后评估(与专业学位 0851-11 内容相同)	306
09	人居科学导论(与专业学位 0851-12 内容相同)	308
10	建筑与城市设计(Ⅰ)(与专业学位 0851-01 内容相同)	309
11	建筑与城市设计(Ⅱ)(与专业学位 0851-02 内容相同)	310
12	建筑与城市设计(Ⅲ)(与专业学位 0851-S03 内容相同)	312
0814	土木工程一级学科研究生核心课程指南	314
01	有限单元法	314
02	高等钢筋混凝土结构理论	320
03	工程项目管理	325
04	高等土力学	328
05	高等钢结构理论	331
06	高等岩石力学	334
07	给水处理理论与技术	336
08	废水处理技术与工程	338
09	高等水力学	341

01 高级算法设计与分析

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否需要调整本课程指南的建议内容。

算法是计算机科学与技术学科的重要基石之一,是人工智能、大数据、图形图像、分布式系统等领域的重要基础,随着相关领域及应用行业的发展而快速发展。因此,算法设计与分析是从事计算机科学与技术等研究与开发所必须掌握的基本技能。

本课程涉及贪心法、分治法、动态规划、线性规划、图算法、近似算法、随机算法、算法前沿方向等一系列高级算法设计与分析技术和 NP 完全理论,体现出较强的理论性、逻辑性、实践性和前瞻性。

通过本课程的学习,研究生能够学习高级算法理论和算法新思想,培养计算思维模式和算法设计及分析能力,提升利用算法技术解决各领域复杂问题的能力,为从事各领域科学研究与开发工作打下良好的逻辑思维和算法基础。

二、先修课程

结构化程序设计,离散数学,数据结构及算法等。

三、课程目标

本课程应体现算法理论性和逻辑性强、应用范围广、发展速度快等特点,将理论教学与案例研究相结合,理论联系实际,引入前沿课题,建设成为一门基础性、实践性和前瞻性的研究生核心课程。

通过本课程的学习,研究生能够掌握的知识、具备的能力主要体现在以下三个方面:

(1) 深入理解各种高级算法的基本思想、适用条件和设计方法,提升对算法的认识,强化计算思维模式。

(2) 掌握主要的算法分析技术,能够从理论上证明算法的正确性,并对算法的有效性(时间复杂度、空间复杂度、近似比等)进行分析。

(3) 能够利用算法的思维方法和设计技巧,设计算法解决相关领域的实际问题,并实现算法优化,提升利用算法解决实际问题的能力。

四、适用对象

计算机科学与技术学科的硕士研究生和博士研究生。

五、授课方式

运用“问题模型化-求解算法化-效率最优化”的教学模式,讲解经典高级算法,根据案例说明算法基本思想与适用条件。把课堂提问、讨论发言、小组活动等引入到教学中,提高学生的学习积极性与主动性,实现以学生为中心的教学。采用课程讲授与选读最新算法相结合的授课方式,实现基本理论与算法新思想的有机融合,强化计算思维模式。运用 PPT、多媒体以及网络实验平台等多种形式实现电子化教学。

六、课程内容

1. 算法概述

介绍算法的基本概念和分析方法。以计算机科学与技术专业各研究领域中的具体问题为例,讲述算法的重要性和本质。

2. 贪心法和分治法

简要介绍贪心法、分治法的基本思想和适用条件。各学科依托单位可根据研究生的算法基础和学科特色,选取难度恰当的例子阐述贪心法和分治法的应用。

3. 动态规划

简要介绍动态规划求解问题的基本要素和一般步骤。各学科依托单位可根据本校研究生的算法基础和学科特色,选取难度恰当的例子阐述动态规划的应用。

4. 线性规划

简要介绍线性规划模型和标准形。重点讲解单纯形法和对偶理论。应以若干经典问题为例,阐述线性规划的应用。

5. 图算法

简要介绍图基本算法。重点讲解最大流算法、最小费用流算法和二分图最大匹配算法。简要分析最大流和最小割的关系。

6. NP 完全理论

简要介绍计算模型、NP、NP 完全概念。重点讲解图灵归约、多项式归约和问题 NP-难证明。

7. 近似算法

简要介绍常用近似算法设计方法。结合若干典型问题,重点讲述近似算法的设计和近似比分析。

8. 随机算法

简要讲述概率论相关知识和随机算法的分类。结合若干典型问题,重点讲述随机算法的设计与分析。

9. 算法前沿方向

各学科依托单位可根据研究生的算法基础和学科特色,适当讲解部分算法前沿方向,例如局部算法、动态算法、在线算法、大数据数据结构与算法等。

七、考核要求

考核成绩主要包括平时成绩与期末考试成绩,各占总成绩的 50%,学科依托单位可根据学科优势与特色,适当调整比例。平时成绩可包括课外作业、课堂出勤、课堂提问、讨论发言和小组活动等。课外作业主要考查学生利用算法解决实际问题的能力;期末考试主要考查学生对各种高级算法核心思想、适用条件的理解和分析算法有效性的能力。

八、编写成员名单

本课程指南建议稿由王建新(中南大学)主笔,并邀请陈建二(中南大学)、冯启龙(中南大学)等相关专家学者研讨撰写,李昂生(北京航空航天大学)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。

02 高级计算机系统结构

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否需要调整本课程指南的建议内容。

计算机系统结构是计算机科学与技术学科最活跃的研究领域之一。从事计算机系统设计及优化的科研人员需要深入理解与掌握计算机系统设计及优化方法;从事操作系统、编译系统等系统软件设计与开发的科研人员需要具有深入理解计算机系统的的功能;从事算法设计与应用的人员需要深刻理解计算机系统,才能写出速度更快、精度更高的应用程序。

本课程涉及并行与分布式计算、互连与通信、数据高效访问和系统量化评价等计算机系统相关的基础知识、最新进展和科研方法,体现出计算机系统结构的系统性、实践性等特点。

通过本课程的学习,研究生能够学会从总体结构和系统分析的角度深入理解计算机系统,掌握计算机系统的各种优化设计及量化评价方法;能够理解计算机系统与系统软件的交互关系;能够面向不同需求,对计算机系统的各个优化目标进行统筹折中,提升自上而下地分析和解决计算机系统相关问题的能力。

二、先修课程

计算机组成原理,计算机系统结构,操作系统,编译原理,计算机编程语言等。

三、课程目标

本课程应体现计算机系统结构从系统角度研究软件/硬件功能、分配和确定软件/硬件界面等特点,建设成为一门具有系统性、实践性、前沿性的研究生核心课程,培养研究生的系统思维,提升研究生对计算机系统的深入理解、优化设计、量化评价的能力。

通过本课程的学习,研究生能够掌握的知识、具备的能力主要体现在以下三个方面。

(1) 学会从总体结构和系统分析的角度深入理解计算机系统,掌握计算机系统相关的基础知识、优化设计与量化评价方法、科研方法。

(2) 能够理解计算机系统与系统软件的交互关系,学会从外部研究计算机系统,培养系统思维和辩证思维能力。

(3) 能够面向不同需求,对计算机系统的各个优化目标进行统筹折中,提升解决计算机系统实际问题的能力。

四、适用对象

计算机科学与技术学科的硕士研究生和博士研究生。

五、授课方式

采用的教学方式和教学方法主要包括:充分挖掘计算机系统不同层次的并行性等线索,将各个知识点贯穿起来,加强授课的系统性;注意介绍知识点发现者当时的心路历程,让学生站在当时系统设计者的角度来看待相关方法和技术点,体现时代性和实践性;注重强调对于多目标系统优化问题的统筹折中,引导学生全面、多角度考虑问题,学会折中设计,培养辩证思维能力。

建议采用多媒体教学,通过丰富的图片和动画讲解重要知识点,帮助同学深入理解;建议开展研究型教学,把计算机系统领域的科研方法引进教学全过程;建议推荐计算机系统领域国际顶级会议和期刊的最新论文给学生课外阅读,并在课堂进行讨论,使学生及时了解计算机系统领域的最新进展。

六、课程内容

建议课程内容以并行性的不同层次为线索展开,涉及单机内部的指令级并行性、线程级并行性和数据级并行性所对应的计算机系统结构和相关软件技术、安全性和性能测量的支撑技术,以及领域专用的系统结构等;多机并行性相关的计算机系统结构,建议涉及多机并行高性能计算系统、互连网络 and 大规模存储系统的关键支撑技术、数据中心体系结构等。各学科依托单位可以根据学科优势与特色,适当调整部分内容。

1. 计算机体系结构与并行性

建议涉及:计算机体系结构定义,性能、功耗度量,器件技术发展趋势以及影响,并行性分类,加速比模型与扩展性分析,安全性、可用性和可测量性的度量等。

■ 重点:区分性能度量中的延迟与吞吐率;功耗与主频的非线性关系是计算机系统结构向并行方式发展的主要原因;程序中的依赖性妨碍并行性的根本原因。

■ 难点:程序中的依赖性。

2. 指令级并行

建议涉及:代码中的指令级并行性,高级分支预测算法和条件访存指令,指令乱序调度方法,基于硬件的推测执行,实例研究(如 Intel i7 处理器核)等。

- 重点:乱序发射的挑战与解决方法;程序行为预测和推测执行思想和实现方式。
- 难点:指令乱序发射,顺序完成的特点。

3. 线程级并行

建议涉及:多线程编程模型与应用,多核处理器体系结构,Cache 一致性模型与实现方法,SMP 与 NUMA 结构,内存一致性模型,内存的多体交叉并行访问,超线程技术,硬件推测执行与事务内存,实例研究(如 Intel Xeon 处理器)等。

- 重点:Cache 一致性模型;内存一致性模型。
- 难点:内存弱一致性模型概念;Cache 一致性与内存一致性的区别和联系。

4. 数据并行

建议涉及:数据并行性与向量化编程,SIMD 指令,编译器自动向量化,SIMT 与 GPU 体系结构,实例研究(如 Intel AVX 指令集,nVidia GPU)等。

- 重点:SIMD 指令的实现;GPU 计算阵列与内存层次。
- 难点:向量化指令的数据对齐和数据重排;GPU 内存层次。

5. 安全性和测量的支撑技术

建议涉及:支持虚拟化的硬件体系结构,安全增强指令(如 Intel SGX),硬件计数器与性能测量。

- 重点:虚拟化的硬件支持。
- 难点:安全增强技术。

6. 领域专用的系统结构

建议涉及:典型领域的计算特征(如深度学习,稀疏矩阵运算),深度学习加速器(如 Google 的 TPU,寒武纪,Microsoft 基于 FPGA 的 Brainwave),面向特定领域的编程语言等。

- 重点:通用体系结构和专用体系结构之间的关系。
- 难点:机器学习加速器的存储布局与数据重用。

7. 多机并行高性能计算机

建议涉及:主要应用与架构,消息传递编程模型,高可用性支持,作业管理等。

8. 互联网络

建议涉及:互联网络拓扑结构与路由算法,多机通信的性能模型,RDMA 技术,实例研究(如 Infiniband 网络等)。

9. 大规模存储系统

建议涉及:RAID,存储区域网络 SAN,分布式文件系统,基于 NVM 的存储系统,实例研究(如 Google 文件系统)等。

- 重点:磁盘故障的检测与恢复。
- 难点:分布式文件系统的一致性。

10. 数据中心体系结构

建议涉及:数据中心的负载特征与编程模型,故障发现、诊断与高可用性,资源分配与任务

调度,云计算模式,高性能计算机与数据中心的比较等。

- 重点:资源隔离机制;分布式系统的高可用性問題。
- 难点:服务在线迁移技术。

建议教学过程中突出重点内容,跟踪国际上系统结构的新发展,及时更新教学内容体系,并采用量化分析方法使学生对计算机系统性能有深刻理解。

七、考核要求

考核主要包括期末考试(占40%)、实验(占40%)和课程论文(占20%)。各学科依托单位可根据研究生质量和学科特色,适当调整各项比例。通过考试,考查学生掌握计算机系统优化的基础知识及量化评价方法。

通过系统实验,考查学生动手实践解决计算机系统实际问题的能力。建议任课教师布置4个实验题目,每个10分;要求学生写明实验的思路,并提交代码。

通过撰写关于最新科研方向的课程论文,考查学生通过综述文献找到研究问题的能力。课程论文可包括综述论文和创新论文两类。综述论文:由教师选定一些题目,同学从中选择,完成综述论文;创新论文:鼓励学生提出新的体系结构思想,并以小论文的形式表达(由于创新论文难度较大,建议给分时考虑附加分)。

八、编写成员名单

本课程指南建议稿由陈文光(清华大学)主笔,并组织张广艳(清华大学)、高鸣宇(清华大学)等相关专家学者研讨撰写,徐志伟(中国科学院计算技术研究所)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。

03 高级分布式系统

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否调整本课程指南的建议内容。

随着信息技术的飞速发展,分布式系统已成为支撑大规模服务的网络化计算机系统,是支撑云计算、大数据等新型计算机应用的计算与存储基础设施,正成为经济社会发展的重要信息基础设施。对于分布式系统的深入理解是研究生从事计算机系统研究与开发的基础,也是从事分布式计算理论与开发的重要基础。

本课程涉及分布式系统简介、分布式通信、并发机制与一致性、分布式共识机制、分布式内

存、分布式文件系统、网格计算系统、P2P 计算系统、云计算系统、分布式数据处理与 MapReduce、分布式数据管理技术、流数据处理技术与系统、数据流技术与系统等分布式系统相关核心内容和最新进展,体现出分布式系统应用范围宽、场景复杂等特点。

通过本课程的学习,研究生能够深入理解与掌握分布式系统及其典型技术特征和支撑技术,能够根据应用场景和需求完成分布式系统的初步设计与优化,为从事分布式计算机系统研究与开发、分布式计算理论与开发等奠定坚实基础。

二、先修课程

计算机网络,计算机操作系统,并行计算基础等。

三、课程目标

本课程应体现分布式系统应用范围宽、场景复杂等特点,建设成为一门具有实践性、前沿性的研究生核心课程,培养研究生的工程思维和系统思维,提升研究生解决分布式系统实际问题的能力。

通过本课程的学习,研究生能够掌握的知识、具备的能力主要体现在以下三个方面:

(1) 深入理解分布式系统的概念、主要原理和分布式系统的典型技术特征,体会分布式系统不同技术指标间的复杂关联关系,为从事分布式系统相关研究与开发工作打好理论基础。

(2) 能够根据应用场景和需求,完成典型分布式系统的初步设计与优化,培养工程思维能力和系统思维能力,提升解决分布式系统实际问题的能力。

(3) 掌握分布式系统在不同阶段的典型支撑技术,了解分布式系统相关的前沿主题,学习研判分布式系统的发展趋势。

四、适用对象

计算机科学与技术学科的硕士研究生和博士研究生。

五、授课方式

建议本课程采用理论与实践相结合、课堂授课与课程项目实践相结合的教学方式,让研究生学以致用、活学活用。建议课堂授课时,辅以课外论文阅读、课堂报告和讨论,提高学生的参与度,激发学生对分布式系统领域的研究与开发兴趣。建议充分利用现代信息技术,采用多媒体教学,利用形象、生动的图片和动画讲解重要知识点,帮助同学深入理解。

建议设置课程实践项目,与目前典型的分布式系统(如云计算系统、大数据处理系统等)相结合,让研究生通过深入接触典型分布式系统,理解其架构思想、技术内涵。

六、课程内容

建议本课程涉及分布式系统基础理论、分布式系统典型支撑技术以及新型分布式系统架构等内容。由于分布式系统应用正处于蓬勃发展阶段,学科依托单位可以根据技术发展现状和学科优势,适当调整教学内容,灵活安排学生实践项目。建议课程内容如下:

1. 分布式系统简介

介绍分布式系统定义、分布式系统模型、分布式系统性能指标、分布式系统 CAP 理论以及分布式系统的发展历史。

2. 分布式通信

介绍分布式系统典型通信技术(多播、RPC、消息通信、流式通信等)。

3. 并发机制与一致性

介绍分布式系统并发控制、分布式一致性以及分布式锁等关键技术。

4. 分布式共识

介绍分布式系统中的容错模型与共识技术、算法以及典型系统(如 Paxos 协议、拜占庭协议、PoW 协议等)。

5. 分布式内存

介绍分布式内存技术背景、关键技术与典型系统。

6. 分布式文件系统

介绍分布式文件系统技术需求、关键技术与实现、技术发展趋势以及典型系统。

7. 网格计算系统

介绍网格计算技术背景、网格体系结构、典型网格计算技术、典型系统与项目。

8. P2P 计算系统

介绍典型 P2P 计算应用需求(文件共享、流媒体等)、支撑技术、典型系统以及最新技术趋势(如区块链)。

9. 云计算系统

介绍系统虚拟化、容器等典型的云计算技术,介绍 IaaS、PaaS、SaaS 云计算模型以及典型云计算系统与应用。

10. 分布式数据处理与 MapReduce

介绍典型大数据处理模式、典型大数据编程模型 MapReduce 以及相关开源系统。

11. 分布式数据管理技术

介绍分布式数据基本概念和核心需求、新型 NoSQL 及 NewSQL 数据库的关键技术与典型系统 HBase 等,介绍分布式数据管理研究现状以及未来发展趋势。

12. 流数据处理技术与系统

介绍分布式流数据处理技术以及典型互联网流数据处理系统与案例。

13. 数据流技术与系统

介绍数据流基本概念、技术思想以及典型数据流处理系统。

上述内容中,分布式系统基础理论(如并发机制与一致性、分布式共识机制)是本课程的重点与难点,应重点讲解。

七、考核要求

考核成绩主要包括学习报告成绩和课程设计成绩,各占 50%。各学科依托单位可根据学科优势与特色,适当调整比例。

建议学习报告由研究生选取一种典型分布式系统技术展开综述,考核研究生对于知识点综述内容的覆盖范围、对于知识点的理解以及对于知识点的逻辑组织。

建议课程设计按照技术发展趋势灵活安排学生实践项目,由研究生完成特定分布式系统的设计,考核研究生对于特定课程设计问题的解决完备程度、性能以及创新思路等,课程设计结业时需提交代码、演示以及设计报告。按照目前分布式技术发展现状,如下课程设计项目可供参考:

(1) 分布式共识系统。基于开源的 Paxos 系统,实现多线程下的 Master 选举功能,比较不同变种 Paxos 的性能(如 Multi Paxos、Fast Paxos、EPaxos 等)。

(2) 分布式数据处理系统。基于分布式节点资源构建批数据处理系统,完成多种类型数据集上的分布式数据处理应用验证(如 Sort、WordCount、PageRank、BFS 等)并实现性能调优。

(3) 云计算系统。基于虚拟化系统搭建云平台,系统具有云计算的 IaaS 平台管理基本功能。

(4) 分布式数据流系统实践。基于 GPU、FPGA 等加速部件,搭建分布式数据流系统,完成机器学习与图计算类迭代应用数据处理,并对比较传统控制流系统性能。

八、编写成员名单

本课程指南建议稿由金海(华中科技大学)主笔,并组织石宣化(华中科技大学)等相关专家学者研讨撰写,郑纬民(清华大学)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。

04 计算机程序理论与模型

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否需要调整本课程指南的建议内容。

计算机程序理论与模型是计算机科学与技术学科及相关领域的重要基础,不仅与计算机系统结构/软件系统的模型及其验证直接相关,而且对其他领域的研究与开发起到重要的支撑作用,是计算机科学与技术学科的研究生需要掌握的重要理论知识。

本课程涉及数理逻辑、程序设计理论中的形式语义、形式语言和并发理论中的计算模型,以及程序验证理论中的模型检测等方面的基础知识,体现出较强的理论性和前瞻性。

通过本课程的学习,研究生应掌握本课程的相关基础理论,强化培养抽象思维和逻辑思维,提高关于计算机程序理论与模型的理论素养,提升发现问题、提出问题、分析问题和解决问题的能力,为从事本学科的科学研究和开发奠定必要的理论基础。

二、先修课程

离散数学,算法设计与分析,数据结构,计算机程序设计等。

三、课程目标

本课程应建设成为一门兼具理论性和前沿性的研究生基础理论课程,让研究生掌握坚实宽广的相关基础理论。通过本课程的学习,研究生能够掌握的知识、具备的能力主要体现在以下三个方面。

- (1) 掌握数理逻辑、形式语义、计算模型、模型检测等计算机科学与技术方面的基本理论和基础知识。
- (2) 拓宽视野,培养抽象思维和严密逻辑推理能力。
- (3) 提升发现问题、提出问题、分析问题和解决问题的能力。

四、适用对象

计算机科学与技术学科的硕士研究生和博士研究生,其他相关学科的研究生。

五、授课方式

建议本课程采用的教学方式方法主要包括:以课堂讲授为主,辅以前沿论文阅读、开源软件工具运用、小组讨论等。

六、课程内容

建议本课程涉及数理逻辑、形式语义、计算模型、模型检测等内容。在具体实施过程中,各学科依托单位可根据学科优势与特色,选择部分内容深入讲授。

第一部分 数理逻辑

1. 命题逻辑

主要包括:命题逻辑公式语法、语义;典型命题逻辑公理系统及其元性质;命题逻辑的可满足性判定一消解法。

重点和难点是公理系统完全性与独立性的证明方法。

2. 谓词逻辑

主要包括:一阶逻辑公式语法、语义;典型一阶逻辑公理系统及其元性质;一阶逻辑公式的消解法及 Herbrand 定理;二阶逻辑的语法、语义。

重点和难点是公理系统完全性与独立性的证明方法。

3. 不完全性定理

主要包括:形式算术、Gödel 数、Gödel 不完全性定理。

重点和难点是 Gödel 不完全性定理的证明。

4. 模态逻辑

主要包括:可能世界模型、模态逻辑语法、模态 μ -演算的语法;表达能力的比较;典型的公理系统及其元性质;各类模态逻辑公式可满足性的判定算法。

重点和难点是 μ -演算的语法、语义、判定算法。

5. 时序逻辑

主要包括:LTL、CTL、CTL* 的语法、语义、表达能力的比较;LTL、CTL、CTL* 的典型公理系统及其元性质;LTL、CTL、CTL* 可满足性判定。

重点和难点是时序逻辑间表达能力的比较、可满足性的可判定性。

第二部分 形式语义

6. 三种基本程序

主要包括:流图型程序、while 型程序、函数式(递归)程序的范型及特点。

重点及难点是 while 型程序及函数式程序的范型。

7. 操作语义

主要包括:流图程序的格局及迁移的定义;while 型程序格局迁移以及计算序列的定义;基于定义计算函数式递归程序的操作语义。

重点及难点是 while 性程序格局迁移以及计算序列的定义。

8. 指称语义

主要包括:完全偏序集、完备格、完备格上的连续函数、连续函数的(最小)不动点; λ -记号及其意义;基于递归语义泛函最小不动点计算程序支撑语义;指称语义与操作语义的等价性。

重点及难点是完备格、连续函数、不动点;指称语义的计算;指称语义与操作语义的等价性。

9. 公理语义

主要包括:程序部分正确性、停机性、完全正确性;程序最弱前置条件、最强后置条件。

重点及难点是程序最弱前置条件(可表达性)、最强后置条件;Hoare 逻辑及其使用。

第三部分 计算模型

10. 形式语言与自动机

主要包括:确定型有穷自动机、非确定型有穷自动机;正则表达式与正则语言;上下文无关文法及上下文无关语言、下推自动机。

重点与难点是文法、自动机、语言之间的等价性;泵引理的使用;各种语言的封闭性。

11. Petri 网

主要包括:网与网系统、Petri 网的动态性质、Petri 网的分析方法、Petri 网的结构性质、Petri 网运算;Petri 网几个重要子类的动态性质分析和判定;颜色 Petri 网、谓词/变迁网系统、增广 Petri 网、时间 Petri 网、随机 Petri 网等扩展 Petri 网定义及性质。

重点与难点是 Petri 网的动态性质分析和判定;顺序、并发、冲突以及同步等关系的理解。

12. 进程代数

主要包括:通信顺序进程(CSP)、通信系统演算(CCS)、 π -演算的语法和形式语义;行为等价理论;公理化及证明系统。

重点及难点是并发性、非确定性;各种行为等价概念;公理化及证明系统; π -演算中的移动性。

第四部分 模型检测

13. 显式模型检测

主要包括:Kripke 结构的构造、基于自动机的 CTL 模型检测方法;基于 tableau 结构的 LTL

模型检测;CTL * 模型检测。

重点及难点是模型检测算法;构建 tableau 结构。

14. 符号模型检测

主要包括:二叉判定图(BDD)及其约简;基于 BDD 的 CTL 符号模型检测;符号模型检测中的公平性;基于 BDD 的 LTL 符号模型检测;基于 SAT 的 LTL 符号模型检测等。

重点及难点是二叉判定图(BDD);符号模型检测算法;公平性。

15. 模型检测应用

主要包括:偏序约简、组合推理;模型检测 Procedural 程序;模型检测并发程序;模型检测安全协议;模型检测概率系统等。

重点及难点是各种模型检测算法及其复杂度问题。

七、考核要求

建议本课程结合研究生的课堂表现(回答问题、论文阅读、小组讨论等),对相关理论与工具的运用、课程论文、期末考试等几方面进行课程考核,形成最终考核成绩。各学科依托单位可根据实际情况,确定上述几方面在最终成绩中所占比例。

八、编写成员名单

本课程指南建议稿由谢冰(北京大学)主笔,并组织王捍贫(北京大学)、曹永知(北京大学)、郭耀(北京大学)等相关专家学者研讨撰写,李宣东(南京大学)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。

05 软件系统与工程

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否需要调整本课程指南的建议内容。

软件系统与工程是计算机科学与技术的重要组成部分,并随着各种新型应用与系统的出现而快速发展,是研究生从事本学科相关领域研究与开发工作的重要基础。

本课程涉及基础软件系统、高级程序设计语言、软件工程等核心内容,为研究生提供较为全面的基础软件和工程技术知识,培养软件思维和工程实践能力。

通过本课程的学习,能够掌握并具备大型、复杂软件系统研究与开发的基础理论知识和工程实践能力,提升计算思维,了解新时代软件系统研究与开发的最新进展,为从事相关领域的研

研究与工程开发工作奠定坚实基础。

二、先修课程

操作系统,数据结构,编译原理,计算机程序设计,数据库,软件工程,计算机网络或分布式系统等。

三、课程目标

本课程应体现软件系统应用范围广、涉及面宽、随各种新型应用与系统的出现而快速发展等特点,建设成为一门系统性、实践性、前瞻性的研究生核心课程。

通过本课程的学习,能够掌握的知识、具备的能力主要体现在以下四个方面。

- (1) 掌握基础软件、高级程序设计语言和软件工程的一般技术和方法。
- (2) 了解新时代软件系统研究与开发的最新进展。
- (3) 理解云计算、大数据等对基础软件系统的要求与影响。
- (4) 掌握并具备大型、复杂软件系统开发和研究的基础理论知识和工程实践能力。

四、适用对象

计算机科学技术学科的硕士研究生和博士研究生,其他相关学科的研究生。

五、授课方式

建议本课程采用课堂讲授、实验实践、课堂研讨等相结合的教学方式。

(1) 课堂讲授。对于基础知识和发展趋势等内容,主要采用课堂讲授的方式进行教学。

(2) 实验实践。对于开发平台和技术,建议采用实验教学方式,在实践中使学生具备切实的实践能力。

(3) 课堂研讨。对于最新研究进展和深入学习需要,采用学生课外阅读相关文献、课堂研讨等方式进行教学。

因课程内容较广泛,建议本课程由多位教师授课,内容上应针对学生情况进行删减;建议以基础知识、原理作为知识点讲授,以实验教学提高学生动手能力。

建议加强从“软件定义”(SDx)的角度,分析软件技术本身的发展趋势和影响。

六、课程内容

建议本课程内容考虑涵盖基础软件系统、高级程序设计语言、软件工程等三部分。各学科依托单位可根据自身优势与特色,适当调整课程内容。

第一部分 基础软件系统

1. 基础软件系统概述

主要包括基础软件系统基本概念,系统软件和基础软件发展历史,主流软件工具/系统栈及其主要示例。

2. 操作系统进阶

主要包括操作系统基础知识回顾;虚拟化与云操作系统;实时操作系统,机器人操作系统的

发展趋势。

3. 数据库系统进阶

主要包括行存与列存;内存数据库管理系统;查询优化,查询的并发执行;数据密集型计算系统,如 SQL、NoSQL、NewSQL 概念;部分系统实例介绍。

4. 中间件与云计算

主要包括传统中间件系统及其架构,云计算基础设施,事务处理机制,区块链系统,边缘计算概念。

第二部分 高级程序设计语言

5. 程序设计语言发展、分类及其设计原理

6. 过程式程序设计语言、面向对象程序设计语言

7. 函数式程序设计语言及其实践

8. 逻辑程序设计语言及其实践

9. 并发程序设计语言与描述性程序设计语言

第三部分 软件工程部分

10. 软件工程概论

主要包括软件工程历史、软件开发范型、软件生存周期模型。

11. 软件项目管理

主要包括项目组织立项、估算与计划、执行与监控;软件工程度量和质量保障体系,如 GQM 和 GQM+度量体系、过程度量与产品度量、ISO9000 质量保证体系和 CMMI 过程改进。

12. 开源软件开发方法

开源协议简介,大教堂与市集、Linus 定律等规律,开源平台与工具简介。

13. 需求工程

主要包括软件需求描述,需求工程典型方法(包括面向目标的方法、面向主体的方法、面向情景的方法、问题驱动的方法等),非功能性需求分析技术,需求规约和验证技术,领域工程和基于领域建模的需求开发技术。

14. 软件体系结构

主要包括软件体系结构的内涵、典型软件体系结构风格;体系结构描述语言 ADL;软件设计模式;软件产品线工程;面向特定领域的软件体系结构介绍。

15. 程序分析与软件测试

主要包括软件质量模型、质量度量;软件测试的理论、方法和测试用例设计;程序静态分析的技术和工具;程序动态分析的技术和工具;软件测试技术和工具、自动化测试技术和框架;软件测试管理工具、缺陷跟踪系统介绍。

16. 模型驱动的软件开发方法

主要包括模型驱动导论,模型驱动工程技术架构,元模型理论、元建模标准、元-元模型体系、模型转换理论和模型转换标准等。

17. 敏捷软件开发方法

主要包括敏捷开发的原理和实践,以及敏捷开发过程的管理和协调等。

18. 软件复用与软件自动化

主要包括软件复用概念、历史和发展趋势,软件开发中的智能推荐技术,领域特定的软件自动化技术。

建议本课程针对一些具体内容设计实验环节,由研究生选择部分实验开展实践,培养动手实践能力。

七、考核要求

建议本课程考核成绩由期末考试、实验考核和课程大报告等三部分构成。各学科依托单位可根据实际情况,适当调整各部分比例。

(1) 书面考核(占总成绩 30%)。对基本概念、原理进行书面考查。

(2) 实验考核(占总成绩 40%)。按照所设置的课程实验的结果进行考核。

(3) 课程大报告(占总成绩 30%)。研究生自由选择感兴趣方向,进行论文研读和总结,形成技术综述报告。

八、编写成员名单

本课程指南建议稿由谢冰(北京大学)主笔,并组织孙艳春(北京大学)、胡振江(北京大学)、郭耀(北京大学)、金芝(北京大学)、黄罡(北京大学)等相关专家学者研讨撰写,李宣东(南京大学)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。

06 数据科学与工程

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否需要调整本课程指南的建议内容。

当前,大数据已经成为一种新的国家战略资源,具有海量数据规模、快速数据流转、多样化数据类型和低价值密度等重要特征,在数据获取、存储、管理、分析及数据保护等方面催生了新的科学理论和工程方法,逐步形成了数据科学与大数据技术,并广泛应用到计算机科学与技术学科的众多领域。对数据科学与工程的深入理解是研究生从事数据驱动的科学研究和技术研发的基础,也是从事数据管理与分析处理相关研究与开发的重要基础。

本课程涉及数据科学与大数据概论、数据采集与预处理、数据管理、数据分析、数据可视化、数据安全与隐私保护、面向大数据处理的分布式计算平台与系统及典型数据处理方法等相关核心内容和最新研究进展,以及大数据典型应用领域的需求特征、数据特征与数据处理的特殊方

法,也可以涉及数据开放共享的法律法规和大数据伦理的一般原则,能够体现出大数据多学科融合、多行业辐射、实践性强等特点。

通过本课程的学习,能够深入理解数据科学的范畴及发展历史,掌握大数据的基本原理、基本知识和基本方法,培养大数据思维,了解数据开放与共享的法律政策规范及大数据应用的基本伦理原则,初步具备开展大数据相关领域科学研究和应用开发的基本能力。

二、先修课程

概率统计,计算机科学导论,程序设计语言,数据库导论等。

三、课程目标

本课程应体现数据科学与工程的多学科融合、多行业辐射、实践性强等特点,建设成为一门具有交叉性、实践性、前沿性的研究生核心课程,培养研究生的大数据思维,提升研究生应用数据方法和思维、开展大数据相关领域科学研究和应用开发等能力。

通过本课程的学习,能够掌握的知识、具备的能力主要体现在以下三个方面。

(1) 深入理解数据科学的范畴及发展历史,掌握大数据的基本原理和基本知识,了解数据安全与隐私保护、大数据平台的关键技术研究进展等。

(2) 深入理解大数据作为一种跨学科基本方法,如何在典型应用领域发挥作用,初步具备应用数据科学的思想方法进行科学研究和解决实际问题的能力。

(3) 了解数据开放与共享的法律政策、规范和一般方法,明确大数据应用的一般伦理原则,对开展数据科学应用过程中的法律和伦理边界建立概念。

四、适用对象

计算机科学与技术学科的硕士研究生和博士研究生。

五、授课方式

建议本课程采用理论与实践相结合、课堂授课与课程项目实践相结合的教学方式,让研究生学以致用、活学活用。建议课堂授课时,辅以课外论文阅读、课堂报告和讨论,提高学生的参与度,激发学生对数据科学与工程领域的研究和开发兴趣。建议学科依托单位结合自身优势和特色,选择性设置应用案例类课程模块,并从行业需求、典型应用、交叉融合等角度,由任课教师或分别邀请境内外、校内外的著名专家学者讲授。有条件的依托单位,应结合自身行业特色和学科优势,设置与应用领域结合的课程实践项目,充分利用互联网、慕课、多媒体等平台,开展实际案例教学及大数据技术能力的实践实训,鼓励与引领行业发展的著名企业合作,到企业研发现场探讨迫切需要解决的关键问题。

授课过程建议将专家学者讲授、研究生调研分析、师生研讨辩论、分组实践等方式结合起来,帮助学生培养综合利用数据方法开展科学研究和解决实际问题的基本能力和基础素养。

六、课程内容

建议本课程涉及数据科学与大数据概论、数据采集与预处理、数据管理、数据分析、数据可

视化、数据安全与隐私保护、面向大数据处理的分布式计算平台与系统及典型数据处理方法等相关核心内容和最新研究进展,以及大数据典型应用领域的需求特征、数据特征及数据处理的特殊方法。

由于数据科学与工程仍处于快速发展阶段,学科依托单位可根据技术发展现状和学科优势,适当调整教学内容,灵活安排学生实践项目。课程内容可以考虑:

1. 概述

介绍数据科学的范畴及对学科发展的影响,以及大数据的发展历程、内涵和外延、价值和意义、技术挑战等。

2. 数据采集与预处理

大数据的多源采集方式、数据集成的一般方法、用于数据变换和数据质量的预处理方法等。

3. 数据管理

关系数据库与结构化查询语言、分布式文件系统、NoSQL 数据库及 SQL on Hadoop 等新型数据管理与查询系统的工作原理及一般查询方法等。

4. 数据分析

用于数据描述及回归分析的统计数据分析方法,基于监督学习及非监督学习方法的数据分析,图数据的分析方法,以及面向自然语言处理的数据分析方法等。

5. 数据可视化

数据可视化的基本原理,面向高维数据、网络数据、层次结构数据、时空数据及文本数据的可视化方法及其高可扩展可视化技术,了解基本的可视化工具和软件等。

6. 数据安全与隐私保护

数据安全与隐私保护的基本概念、支撑技术、数据交互安全与数据脱敏、数据生命周期安全的防护及管理体系等。

7. 大数据处理平台

大数据处理平台架构,面向批量数据、流式数据、图数据的处理方法及并行优化方法等。

8. 大数据应用

从社会网络大数据、城市大数据、工业大数据、教育大数据等方面选取案例进行剖析,体现不同应用领域的需求特征、数据特征及数据处理的特殊方法,满足多学科交叉融合的能力需求。

9. 大数据治理

介绍数据开放与共享过程中的一般实践,以及可能涉及的法律法规等;通过案例介绍大数据伦理的一般原则,明确基于数据方法开展应用的基本法律和伦理边界等。

建议本课程讲授 32—48 课时。依托单位可根据实际情况,选择 1—2 个不同行业的应用案例进行重点讲解,并按需设置实验实训教学任务。

七、考核要求

建议本课程考核成绩主要包括平时成绩与期末成绩,建议平时成绩占比不超过 50%,各学科依托单位可根据学科优势与特色,适当调整比例。平时成绩可以涉及课堂听讲、调研报告、讨论辩论等;期末成绩可以根据课程内容的知识点掌握情况进行考查,或通过课程项目、实践报告等方式进行综合衡量。

建议本课程考核标准主要考虑:是否了解数据科学的范畴及对学科发展的影响,掌握大数据应用生命周期各个环节的基本原理、基础知识和一般方法;是否初步具备应用数据科学的思想方法进行科学研究和解决实际问题的能力;是否了解数据开放与共享的法律政策、规范和一般方法,明确大数据应用的一般伦理原则,并对开展数据科学应用过程中的法律和伦理边界建立概念。

八、编写成员名单

本课程指南建议稿由胡春明(北京航空航天大学)主笔,并组织吕卫锋(北京航空航天大学)、马帅(北京航空航天大学)、吴中海(北京大学)、王宏志(哈尔滨工业大学)、王建民(清华大学)、袁晓如(北京大学)、唐杰(清华大学)、陈恩红(中国科学技术大学)、刘闯(中国科学院地理科学与资源研究所)、马民虎(西安交通大学)等相关专家学者研讨撰写,杜小勇(中国人民大学)、王国仁(北京理工大学)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。

07 高级计算机网络

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否需要调整本课程指南的建议内容。

作为计算机科学与技术学科及相关领域的重要载体,互联网经过 50 年的发展,已成为陆海空天之后的人类第五疆域“网络空间”,是推动人类社会发展的巨大动力。

建议本课程在本科生计算机网络课程的基础上,进一步深入探讨计算机网络研究和应用领域中的一些高新技术、前沿技术以及建模方法与仿真工具,体现计算机网络的系统性、前沿性和实践性。

通过本课程的学习,能够深入理解计算机网络体系结构及相关高新技术和前沿技术,掌握部分网络建模方法和仿真工具,提高对计算机网络与分析、设计和理解能力;能够了解计算机网络前沿技术领域及最新进展,体会计算机网络领域的发展趋势,培养创新思维能力,为从事计算机网络领域的学习、研究与开发工作打下坚实的基础。

二、先修课程

计算机网络,程序设计语言等。

三、课程目标

本课程应体现计算机网络涉及面宽、发展快、应用场景广等特点,主要围绕计算机网络高级主题、前沿技术、建模方法与仿真工具等内容开展,建设成为一门系统性、前沿性、实践性的研究生核心课程。

通过本课程的学习,能够掌握的知识、具备的能力主要体现在以下三个方面:

(1) 深入理解计算机网络体系结构的概念及各项高级技术的基本原理和方法,提高对计算机网络及其应用的分析、设计和理解能力。

(2) 了解计算机网络前沿技术领域及其最新发展现状,体会计算机网络领域的发展趋势,培养创新思维能力。

(3) 掌握部分网络仿真技术和仿真工具,为从事计算机网络领域的学习、研究与开发工作打下基础。

四、适用对象

计算机科学与技术学科的硕士研究生和博士研究生,其他相关学科的研究生。

五、授课方式

建议本课程采用以下授课方式:课堂讲授与师生研讨相结合、课堂教学与学生自学相结合、课堂知识与课外阅读相结合、理论学习与动手实践相结合。

六、课程内容

建议本课程涉及计算机网络高级主题、计算机网络前沿技术、计算机网络建模方法与仿真工具三部分内容。

第一部分 计算机网络高级主题

1. 网络体系结构概述

建议涉及:网络体系结构的基本概念、主要内容和设计原则,OSI 体系结构和 TCP/IP 体系结构,网络体系结构的演进与面临的挑战。

2. 网络路由交换技术

建议涉及:网络命名与寻址技术(IP 地址分配方法、IPv4/IPv6 共存技术、移动 IP 技术),网络交换技术(第三层交换模型、IP over SDH 技术、IP over WDM 技术、MPLS 技术),网络路由技术(基本概念和原则、单播路由算法与协议、多播路由算法与协议)。

3. 网络拥塞控制与 QoS

建议涉及:拥塞控制的基本概念,网络流与主动队列管理,TCP 拥塞控制机制,显示拥塞告知机制(ECN),用户体验质量(QoS)的基本概念,QoS 模型与评价方法。

4. 无线网络

建议涉及:无线网络体系结构,无线局域网(WiFi),无线个域网(蓝牙、Zigbee),无线城域网(WiMax),无线 Mesh 网络,移动 Ad Hoc 网络,移动机会网络。

5. 网络安全

建议涉及:网络安全体系,网络安全协议,网络态势感知,匿名通信及检测,隐蔽通信及检测,移动网络安全接入,数据鉴别,入侵检测与防范,网络测量与流量分析。

第二部分 计算机网络前沿技术

6. 软件定义网络(SDN)与网络功能虚拟化(NFV)

建议涉及:SDN 基本思想与架构,SDN 交换机与控制器,SDN 南向接口与北向接口,OpenFlow/NetConf 协议,网络功能虚拟化(NFV),P4 网络编程语言,软件定义广域网络(SD-WAN)。

7. 信息中心网络(ICN)

建议涉及:ICN 基本思想与体系架构,对等网络(P2P),覆盖网络(Overlay Network),内容分发网(CDN),内容中心网络(CCN),NDN 体系结构(命名、路由等),ICN 的发展与应用。

8. 数据中心网络(DCN)

建议涉及:数据中心的概念,数据中心网络拓扑结构,数据中心网络互联,数据中心网络路由与虚拟化,数据中心网络流量协同传输,数据中心网络安全,SDN 与数据中心网络,云计算与数据中心网络,数据中心网络的发展与应用。

9. 物联网(IoT)

建议涉及:物联网概念与体系结构,物联网感知技术,无线传感网,无线定位,物联网数据融合,群智感知与边缘计算,物联网信息安全与隐私保护,物联网的发展与应用。

第三部分 计算机网络建模方法与仿真工具

10. 排队论与网络性能分析

建议涉及:概率论与随机过程,泊松分布,马尔可夫排队模型,非马尔可夫排队模型,网络排队模型,自相似理论与模型,网络流量建模与分析方法。

11. 协议工程与形式化方法

建议涉及:协议工程概述,网络协议形式描述技术(FDT),网络协议验证方法,网络协议实现和测试方法。

12. 网络仿真技术与工具

建议涉及:网络仿真技术的基本概念,OPNET、NS、GloMoSim、Mininet、PlanetLab 等网络仿真工具和平台。

授课单位可结合自身特色与优势,适当增减上述课程内容。

对于 2 学分的课程,建议主要授课内容为第 1~8 部分。

对于 3 学分的课程,建议在 2 学分主要授课内容的基础上,增加第 7、12 部分。

对于 4 学分的课程,建议在 3 学分主要授课内容的基础上,增加第 10、11 部分。

七、考核要求

建议本课程考核成绩包括平时成绩与期末成绩。平时成绩包含课堂研讨和动手实践,各占总成绩的 25%;期末成绩占总成绩的 50%。学科依托单位可根据学科优势与特色,适当调整各部分比例。

八、编写成员名单

本课程指南建议稿由罗军舟(东南大学)主笔,并组织苏金树(国防科技大学)、徐明伟(清华大学)、李伟(东南大学)等相关专家学者研讨撰写,罗洪斌(北京航空航天大学)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。

08 网络与信息安全

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据其优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否需要调整本课程指南的建议内容。

网络与信息的安全是国家安全的重要组成部分,也是计算机科学与技术学科的核心内容之一,涉及计算机硬件、软件、系统与应用等各个领域。计算机科学与技术学科的研究生应了解和掌握相关知识,牢固树立网络与信息安全观并付诸实践,保障国家网络空间安全。

建议本课程围绕网络与信息的安全问题,涵盖相关的主流理论与技术,可以涉及密码理论与技术、计算机硬件安全、软件安全、系统安全和应用安全等,体现网络与信息安全的综合性、复杂性、实践性和前沿性。

通过本课程的学习,应系统地掌握网络与信息安全领域的核心技术,牢固树立网络与信息的安全观,提升在网络与信息安全领域的科研能力、创新能力和实践能力,为从事网络与信息安全的研究和开发工作打下坚实的基础。

二、先修课程

程序设计,操作系统原理,计算机网络等。

三、课程目标

本课程应针对网络与信息的安全问题,体现出涉及面广、综合性强、复杂性高、实践性强等特点,建议考虑涵盖密码理论与技术、计算机硬件安全、软件安全、系统安全和应用安全等,建设成为一门具有综合性、实践性、前沿性的研究生核心课程。通过本课程的学习,能够掌握的知识、具备的能力主要体现在以下几个方面:

(1) 深入学习网络与信息安全领域的基础理论和前沿技术,系统地掌握设计和开发网络与信息安全的基本方法和高级技术。

(2) 能够自己动手解决与网络与信息安全相关的技术难题和实际问题,提升在网络与信息

安全领域的科研能力、创新能力和实践能力。

(3) 牢固树立网络与信息安全意识,具备将其融入计算机科学与技术相关领域科学研究与开发工作的能力。

四、适用对象

计算机科学与技术学科的硕士研究生和博士研究生,其他相关学科的研究生。

五、授课方式

建议本课程采用课堂授课和实验相结合的教学方式。课堂授课提倡师生课堂互动,建议每次课预留 15 分钟的师生互动时间,针对特定网络与信息安全问题及其解决方案进行研讨。

建议课外实验由任课教师布置若干网络与信息安全的实验,研究生结合研究方向和兴趣,选做其中一个,提高动手实践能力。

六、课程内容

建议本课程考虑涵盖网络与信息安全相关的主流理论与技术,可以涉及密码技术、对称加密算法、消息摘要与消息认证码、非对称密码算法、公钥基础设施、用户认证机制、Internet 安全协议、无线移动安全、防火墙与虚拟专用网(VPN)、软件安全、操作系统安全、可信计算、移动智能终端系统安全和物联网(IoT)安全等。各学科依托单位可根据单位优势与特色,适当增减内容。

1. 网络与信息安全概述

建议涵盖以下内容:安全需求、安全方法、安全性原则、攻击类型等。重点、难点是安全性原则、攻击类型。

2. 密码技术

建议涵盖以下内容:明文与密文、替换加密技术、变换加密技术、加密与解密、对称与非对称密钥加密、密码分析及攻击等。重点、难点是对称与非对称密钥加密、密码分析及攻击。

3. 对称加密算法

建议涵盖以下内容:算法类型与模式、数据加密标准(DES)、国际数据加密算法(IDEA)、RC4、高级加密标准(AES)、中国商用密码算法 SM4 等。重点、难点是 DES、AES、SM4。

4. 消息摘要与消息认证码

建议涵盖以下内容:消息摘要、MD5、安全散列算法、SHA-512、SHA-3、消息认证码、HMAC、中国商用密码算法 SM3。重点、难点是 MD5、SHA-512、HMAC、SM3。

5. 非对称密码算法

建议涵盖以下内容:非对称密钥加密简史、RSA 加密算法、ElGamal 加密算法、RSA 签名算法、ElGamal 签名算法、中国商用密码算法 SM2、Diffie-Hellman 密钥交换。重点、难点是 RSA 加密算法、ElGamal 加密算法、ElGamal 签名算法、SM2、Diffie-Hellman 密钥交换。

6. 公钥基础设施

建议涵盖以下内容:数字证书、私钥管理、PKIX 模型、公钥加密标准、PKI 与安全。重点、难点是数字证书、PKIX 模型、公钥加密标准。

7. 用户认证机制

建议涵盖以下内容:认证基础、口令、认证令牌、基于证书认证、生物认证、Kerberos、密钥分发中心、单点登录方法(Single sign-on,SSO)。重点、难点是口令、基于证书认证、Kerberos。

8. Internet 安全协议

建议涵盖以下内容:IPSec、安全套接层(Secure Socket Layer,SSL)、传输层安全(Transport Layer Security,TLS)、安全超文本传输协议(HTTPS)、安全电子事务规范(SET)、电子邮件安全。重点、难点是IPSec、TLS、SET。

9. 无线移动安全

建议涵盖以下内容:WAPI、802.11i、GSM 安全、3G/4G/5G 安全、无线应用程序协议安全。重点、难点是WAPI、802.11i、3G 安全。

10. 防火墙与 VPN

建议涵盖以下内容:防火墙的类型、防火墙配置、防火墙的局限性、入侵检测类型、分布式入侵检测、Honeypot 技术、虚拟专用网。重点、难点是防火墙的类型、分布式入侵检测、虚拟专用网。

11. 软件安全

建议涵盖以下内容:软件缺陷、恶意软件、僵尸网络、软件逆向工程。重点、难点是软件缺陷、软件逆向工程。

12. 操作系统安全

建议涵盖以下内容:隔离机制、内存保护、访问控制、多级安全模型。重点、难点是访问控制、多级安全模型。

13. 可信计算

建议涵盖以下内容:可信计算的定义与原理、可信根和可信链、可信密码服务、可信部件、可信体系架构。重点、难点是可信根和可信链、可信密码服务。

14. 移动智能终端系统安全

建议涵盖以下内容:移动智能终端系统面临的安全威胁、恶意移动应用检测与防御、移动用户数据隐私保护、Android 系统安全特性和应用安全模型。重点、难点是恶意移动应用检测与防御、移动用户数据隐私保护、Android 系统安全特性和应用安全模型。

15. IoT 安全

建议涵盖以下内容:RFID 标签攻击和防御、RFID 标签指纹、SIM 卡安全、智能汽车安全、智能摄像头安全。重点、难点是 RFID 标签攻击和防御。

建议本课程安排以下实验,研究生可结合研究方向和兴趣,选做一个。各学科依托单位可根据学科特色与优势,设置其他相关实验。

实验 1:利用 IPsec 构建 VPN。

实验 2:基于 Android 系统,实现基于 Hook 机制的用户隐私数据保护。

实验 3:利用 RFID 读写设备,实现回放攻击。

七、考核要求

建议本课程考核成绩由实验成绩和期末考试成绩组成。

实验成绩:40%。学生任选一个实验题目,实现该题目所要求的功能并提交实验报告。主

要考核学生的网络与信息安全设计与实现能力,运用所学知识分析并解决具体问题的能力。

期末考试成绩:60%。采用书面闭卷考试形式,主要考核学生对网络与信息安全相关基础知识和高级技术的掌握程度。

各学科依托单位可根据学科优势与特色,适当调整上述比例。

八、编写成员名单

本课程指南建议稿由马建峰(西安电子科技大学)主笔,并组织姜奇(西安电子科技大学)等相关专家学者研讨撰写,李兴华(西安电子科技大学)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。

09 机器学习

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否需要调整本课程指南的建议内容。

人工智能是本学科的重要研究方向之一,是引领未来的战略性技术和新一轮产业变革的核心驱动力;我国将发展人工智能技术与产业作为国家重大战略。而机器学习是人工智能的核心领域之一,在计算机各个领域快速渗透,对计算机科学与技术学科的各个领域均有影响和支持作用。

建议本课程考虑涉及假设空间、归纳偏好、模型评估选择、包括线性模型、决策树、神经网络、支持向量机、贝叶斯分类等在内的常用机器学习方法,以及集成学习、聚类、降维及度量学习、特征选择、稀疏学习、强化学习等内容,体现基础性和学科交叉性。

通过本课程的学习,能够深入理解机器学习原理,掌握常用机器学习方法和技术;能够结合具体研究领域,针对实际问题提出基于机器学习技术的解决方案,提升实践与创新能力。

二、先修课程

数据结构与算法,概率论与数理统计,线性代数,数值计算等。

三、课程目标

本课程应体现机器学习基础性强、应用领域宽等特点,应建设成为一门具备基础性和交叉性的研究生核心课程。

通过本课程的学习,能够掌握的知识、具备的能力主要体现在以下三个方面。

(1) 较全面地了解机器学习方法,理解机器学习原理,掌握常用机器学习方法和建模技术,能够灵活运用多种技术手段,进而提炼、丰富相关知识结构。

(2) 能够使用机器学习方法解决特定问题,基本具备对实际问题进行建模的能力。

(3) 对学有余力或有志于投身机器学习研究的学生,能够结合实际问题的特性,有针对性地提出基于机器学习技术的建模与解决方案,提升实践能力与创新能力。

四、适用对象

计算机科学与技术学科的硕士研究生和博士研究生,其他相关学科的研究生。

五、授课方式

建议本课程的授课方式为:以课堂讲授为主,结合实践作业/项目和课程研讨。

建议本课程充分利用 MOOC 等网络资源,将课堂授课由全面讲解,转为课前自学、课堂重点讲解和疑问解答相结合,提高授课效率。

建议本课程安排扩展研讨,在稳固基础的同时,给予学生更为广阔的视角。对于小课堂教学(如修课人数少于 20 人),可通过师生互动,探讨机器学习领域的新进展;对于大课堂教学,可通过课前分组、课前预研、课堂选举演讲的方式开展研讨,释放学生主观能动性,提升学习兴趣。

六、课程内容

建议本课程考虑涉及机器学习概论、模型评估与选择、线性模型、决策树、神经网络、支持向量机、贝叶斯分类、集成学习、聚类、降维与度量学习、特征选择与稀疏学习、强化学习等内容。各学科依托单位可根据学科优势与特色,结合实际需求适当调整。

1. 机器学习概论

建议介绍机器学习的基础概念、发展历程和应用现状,重点讲述假设空间、归纳偏好、没有免费午餐定理等基本原理知识。

2. 模型评估与选择

建议重点介绍经验误差与过拟合、评估方法(交叉验证的方法),介绍性能度量和比较检验,让学生理解偏差、方差、噪声等概念。

3. 线性模型

建议介绍线性模型基本形式,重点讲述线性回归、对数概率回归、线性判别分析等经典方法。

4. 决策树

建议介绍决策树基本思路和流程,重点讲述三类重要划分选择(信息增益、增益率、基尼指数)及其重要代表性方法(ID3 决策树、C4.5 决策树、CART 决策树),介绍剪枝处理方法。根据具体需要,可进一步剖析决策树方法的优势和缺点。

5. 神经网络

建议介绍神经元模型、感知机模型等基本知识,重点讲述误差逆传播算法(BP 算法)以及常见神经网络。根据具体需要,可介绍正则化、模拟退火、随机梯度下降等优化技术和深度学习重要前沿结果等。

6. 支持向量机

建议介绍间隔和支持向量等基本概念,重点讲述支持向量机模型推导,并介绍对偶问题、核函数、正则化、软间隔等技术的作用和原理。根据具体需要,可进一步剖析支持向量机的优势和缺点。

7. 贝叶斯分类

建议介绍贝叶斯决策论,重点介绍极大似然估计方法、贝叶斯分类器(朴素/半朴素贝叶斯、ODE、贝叶斯网);介绍拉普拉斯修正的作用,重点介绍 EM 算法。

8. 集成学习

建议重点介绍 Boosting、Bagging、随机森林等经典方法,剖析集成学习的结合策略(平均法、投票法、学习法),引出集成学习在统计、计算、表示三个方面的直观意义和原理;最后可以根据具体情况,介绍多样性增强对集成学习的重要性。

9. 聚类

建议介绍聚类任务的背景,重点讲述聚类性能度量的意义,包括外部指标和内部指标,讲述原型/密度/层次聚类等聚类做法的原理、意义、和适用范围。

10. 降维与度量学习

建议简要介绍 k-近邻学习,重点讲述低维嵌入、主成分分析、核化线性降维等重要降维方法;介绍流形学习、等度量映射等高级降维技术,重点讲述度量学习作用和意义。

11. 特征选择与稀疏学习

建议简要介绍子集搜索和评价,重点讲述过滤式/包裹式/嵌入式等三种重要特征选择方法的原理、优缺点和代表性技术、阐述稀疏表示与 L1 正则化。根据实际情况,可进一步介绍字典学习、压缩感知等高级技术。

12. 强化学习

建议介绍任务和奖赏的概念,以多臂赌博机为例,讲述探索-利用原理在强化学习中如何体现,重点介绍有(免)模型学习、模仿学习等技术原理。

建议本课程安排实践作业或实践项目(例如使用 Python、Tensorflow、Pytorch 等),提高学生的机器学习实践能力。实践内容可以根据学生的平均基础,选择 2~5 个知识点,由老师提供相应的数据、机器学习任务目标以及机器学习基本编程框架,安排学生完成机器学习算法编程实践并撰写技术报告。

七、考核要求

建议本课程考核成绩主要包括期末笔试成绩与平时实践成绩,各占总成绩的 50%。各学科依托单位可根据学科优势与特色,适当调整比例。

期末笔试主要考查学生对机器学习方法、原理和建模技术的掌握情况。建议期末笔试成绩的考核标准为:基本概念了解题,30%;算法原理理解题,30%;机器学习理论基础题,10%;算法使用和计算题,20%;机器学习应用能力题,10%。

平时实践主要考查学生对实际问题进行建模的能力与表达能力。建议平时实践成绩的考核标准为:完成算法实现,20%;完成机器学习任务,30%;完成算法性能分析,30%;报告撰写规范,20%。

八、编写成员名单

本课程指南建议稿由周志华(南京大学)、詹德川(南京大学)和李宇峰(南京大学)等撰写,于剑(北京交通大学)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。

10 学科前沿与实践

一、课程概述

推荐本课程作为计算机科学与技术一级学科的研究生核心课程。建议有关学科依托单位根据优势和特色,自行决定是否将本课程纳入培养方案的核心课程体系,自行决定是否需要调整本课程指南的建议内容。

计算机科学与技术具有发展速度较快、涉及方向较宽、行业需求较大等特点,学科依托单位应结合本单位具有的优势和特色,将本课程建设成为一门前沿性、交叉性、实践性的研究生核心课程。

本课程一般应由系列课程模块组成,研究生根据研究方向、职业规划、兴趣爱好等因素,在导师的指导下选择其中的若干课程模块作为学习实践任务。

通过本课程的学习,能够了解计算机科学与技术国际前沿的新动态、新概念和新热点,在及时掌握新知识点的同时,也培养了针对学科发展的敏感性。一般地,其他核心课程难以覆盖这些国际前沿的新知识点,各类新知识点还不足以分别构成相对独立的新课程。

通过本课程的学习,能够将国际前沿的新知识点与实际面临的关键问题结合起来,从科学研究、技术开发、行业应用等方面选择一个探索点,动手完成一个实践任务。当国际前沿的新热点成为未来的关键点时,本课程能够成为研究生未来具有竞争优势的起点。

通过本课程的学习,能够立足多个方向的国际前沿交叉点,既充实了多个方向的新知识点,感受了学科方向之间的相互关联,又体验了不同方向之间的相互启发作用,训练了交叉型的创新思维能力,也为培养本学科的系统设计能力奠定了基础。

二、先修课程

计算机科学与技术学科的本科生相关基础课程和专业课程。学科依托单位根据本课程的模块组成,确定其先修课程涉及的本科生相关基础课程和专业课程。

三、课程目标

本课程应体现计算机科学与技术学科发展速度较快、涉及方向较宽、行业需求较大等特点,

应体现学科依托单位具有的优势和特色,应建设成为一门前沿性、交叉性、实践性的研究生核心课程。

研究生根据研究方向、职业规划、兴趣爱好等因素,在导师的指导下选择其中的若干课程模块作为学习实践任务。通过本课程的学习,研究生能够掌握的知识、具备的能力主要体现在以下三个方面。

(1) 了解计算机科学与技术国际前沿的新动态、新概念和新热点,掌握学科发展趋势的调研、分析、判断等方式方法,初步具备关于学科发展敏感性的专业素养。

(2) 将国际前沿的新知识点与实际面临的关键问题结合起来,从科学研究、技术开发、行业应用等方面选择完成一个探索性的实践任务,初步具备将新的知识点转变为行业竞争点的职业能力。

(3) 立足多个方向的国际前沿交叉点,在充实多个方向的新知识点的同时,感受学科方向之间的相互关联,体验不同方向之间的相互启发作用,提升交叉型创新思维能力和系统设计能力。

四、适用对象

计算机科学与技术学科的硕士研究生和博士研究生。

五、授课方式

建议本课程采用的教学方式方法主要包括:针对各个课程模块,从国际前沿、迫切需求、交叉融合等角度,分别邀请境内外、校内外的著名专家学者讲授;有条件的依托单位,可以考虑通过互联网、慕课、多媒体、虚拟现实等平台,与国际一流的高等院校、科研院所、学术机构及会议合作,也可以考虑与引领行业发展的著名企业合作,到企业研发现场探讨迫切需要解决的关键问题。

建议将专家学者讲授、研究生调研分析、师生研讨辩论、分组实践竞赛等方式结合起来,探索以国际前沿的新知识点解决领域面临的关键问题的新思路。

六、课程内容

依托单位选取本课程内容的原则主要有:课程内容的国际学术前沿性、行业需求迫切性、学科方向交叉性;依托单位具有的相关优势和特色,及其课程建设的条件和能力;其他核心课程难以覆盖的国际前沿新知识点,并且这些新知识点还不足以分别构成相对独立的新课程。

根据上述原则,依托单位确定本课程的模块组成,以及每个模块涉及的课程内容,重点可以包括但不限于:每个模块的国际前沿新动态、新概念和新热点,主要为近3年的最新知识点;部分模块应涉及相关行业迫切需要解决的关键问题,其中优先考虑国家发展战略面临的卡脖子问题;多个学科方向关注的共性科学问题、基础理论方法、共性关键技术等新近内容及知识点。

本课程的模块类型可以涉及但不限于:前沿前瞻模块、特色强化模块、交叉融合模块、探索实践模块等。建议本课程包括8—10个模块,每个模块约4—6个讲授课时;每名研究生选择其中的5—6个模块,完成约24个课时的听讲和讨论任务,以及课外调研、讨论、实践等任务。

七、考核要求

建议本课程考核成绩主要包括平时成绩与期末成绩,各占 50%,各学科依托单位可根据学科优势与特色,适当调整比例。平时成绩可以涉及课堂听讲、调研报告、讨论辩论等;期末成绩可以涉及课程项目、实践报告等,部分选题可以鼓励项目演示等。

建议本课程考核标准主要考虑:对相关国际前沿新动态、新概念和新热点的了解程度,以及对学科发展的敏感性;以新的知识点解决关键问题的思维和实践能力;多个方向的交叉融合与相互启发作用,以及课程小组的组织协作能力等。

八、编写成员名单

本课程指南建议稿由马华东(北京邮电大学)主笔,并组织胡斌(兰州大学)、黄华(北京理工大学)等相关专家学者研讨撰写,陈小武(北京航空航天大学)审阅了建议稿,并反馈了修改意见和建议。特别感谢计算机科学与技术学科评议组的审议意见,以及在多次征求意见时收到的高校院所相关学科负责人及专家学者反馈的意见建议。