

《软件系统优化》教学大纲

课程代码	DATA0031132026	课程性质	专业选修
课程名称:	软件系统优化		
英文名称	Optimization for Software Systems		
学时/学分	72/3	其中实验/ 实践学时	36
开课单位	数据科学与工程学院	适用专业:	数据科学与大数据技术专业
先修课程	程序设计、算法基础、计算机系统		
大纲撰写人	郭健美、黄波	大纲审核人	钱卫宁
课程网址	https://www.shuishan.net.cn/education/course/1569130547979730945	授课语言	中文

一、课程说明

如何在给定的硬件系统资源配置下提升软件系统的性能是数字化系统的设计和实现必须思考和解决的问题，同时也是最大化利用硬件平台资源的有效手段。软件系统优化的原理、技术和实践是一位卓越的系统架构师或数据科学家必备的素养，发起软件系统优化方面的课程设置及教学同时也是解决国家计算机系统方面“卡脖子”问题人才培养的有效措施，在培养学生解决实际问题的过程中可以充分培养学生的逻辑思维、批判性思维和创造性思维。

How to improve the performance of software systems with the given hardware resources is a key issue that should be addressed during the design and implementation of digital systems. Elegant approaches to resolving this issue are critical to maximize the utilization of hardware resources. The principles, technologies and practices of software system optimization are essential knowledges to be embraced by an excellent system architect or data scientist. Initiating the curriculum and the teaching of software system optimization undoubtedly facilitates cultivating the talents that are needed to narrow down the gap with leading countries in the domain of computer systems. It also helps improve the students' capability of resolving practical issues, at the same time developing students' logical thinking, critical thinking and creative thinking.

二、课程目标

期望学生通过本课程的学习，达成如下目标：

目标 1：了解应用负载、操作系统、编译器、计算机体系结构的基本概念，掌握软件系统性能优化的基本方法和原理，培养系统观。（支撑毕业要求 7、13、14）

目标 2：掌握程序性能优化的基本原则与方法，掌握编译基本过程和工具框架，学会如何编写高性能代码。（支撑毕业要求 7、9、12）

目标 3：熟练运用系统性能优化的工具和套件，掌握性能数据采集、处理和分析的基本方法和技能，了解相关开源技术社区。（支撑毕业要求 9、12）

目标 4：掌握软件系统性能工程和基准测试的基本概念和方法，了解如何通过实测数据分析和优化方法来定位和解决软件系统的性能瓶颈。（支撑毕业要求 9、12、13、14）

目标 5：了解系统性能优化的演进过程和研究动向，为后续相关课程学习和研究打下良好基础，为后续职业发展奠定基石。（支撑毕业要求 7、13、14）

三、课程目标与毕业要求的对应关系

毕业要求	指标点	课程目标
学科理念先进	7、深刻理解数据的获取、建模、管理、利用的全生命周期，深刻理解数据科学与工程相关技术发展与社会经济发展的关系。	目标1、2、5
工程能力全面	9、掌握主要的数据管理和处理工具以及系统平台的使用，熟知它们的特点、系统架构，具备基本的数据系统的设计和开发能力。	目标2、3、4
	12、具备参与数据系统或数据应用设计、开发、运维工程所需的沟通交流与协作能力，掌握基本的工程管理知识与能力。	
研究能力突出	13、了解“数据科学与工程”学科领域，以及相关应用领域的技术发展前沿。	目标1、4、5
	14、具有初步的从事数据科学与工程研究工作的科学训练，具有从事相关学科科学研究、教学或工程开发的技术工作的能力。	

四、教学内容与学时安排

第一章 绪论（支撑课程目标 1、2、3、4、5）

学时：4

1. 课程介绍

2. 矩阵乘法优化案例

要求学生：通过一个完整案例对软件系统优化课程涉及的概念、方法、工具和效果有个总体了解，开始树立系统观。

实验一：初试 Linux 环境和工具

第二章 应用负载优化（支撑课程目标 1、3、4）

学时：16

1. 应用负载的性能测量
2. 实验设计和参数调优
3. 基准测试程序和方法
4. 负载画像和性能评价

要求学生：了解应用负载的基本概念，掌握应用负载优化相关的基本概念、方法和工具，掌握性能工程的基本知识和技能。

实验二：SPECjvm2008 基准测试

实验三：矩阵乘法的自动调优器

第三章 编译器和编译优化（支撑课程目标 1、2、4）

学时：24

1. 源程序级别的常见优化方法
2. 编译器概论、动/静态编译
3. 目标机器汇编语言
4. 高级程序的汇编代码生成
5. 常见编译优化方法
6. 程序插桩及优化机会识别

要求学生：了解编译器和动/静态编译的基本概念，掌握编译优化相关的基本概念、方法和工具。

实验四：GCC 与 Clang/LLVM 优化比较

实验五：指令向量化 Vectorization

实验六：交叉编译与跨平台应用仿真

第四章 计算机体系结构优化（支撑课程目标 1、2、3、4）

学时：16

1. 计算机体系结构概论
2. 多核异构编程
3. 微架构性能分析方法
4. 高速缓存及相关优化

要求学生：了解计算机体系结构的基本概念，掌握多核异构编程和高速缓存优化相关的基本概念、方法和工具。

实验七：基于 Intel oneAPI 的 CPU+GPU 异构编程

实验八：归并排序的剖析和优化

第五章 前沿研究和应用（支撑课程目标 5）

学时：12

1. 数据中心和云端服务优化
2. 机器学习框架优化

要求学生：了解软件系统优化的前沿研究方向，了解云计算、AI 优化等领域的研究动向。

实验要求：

所有实验包括 5 个上机作业和 3 个实践项目，都需要独立完成，并按时提交电子版报告。作业要求会在“布置周”实践课上发放，作业必须在“提交周”实践课上完成提交，过期将影响成绩。实践项目报告需要包括完整的项目介绍、优化过程描述、算法描述、代码实现和具体的执行过程和结果。上机作业报告主要包括上机执行过程和结果，如有代码实现或其他要求，也要附上。

五、教学方法

本课程注重理论与实践相结合，建立基于项目学习（Project-Based Learning, PBL）的计算机系统能力培养与评价机制，着重加强高年级本科生动手实践能力的培养。首先，课程设计了大量实验，包括 5 个上机作业和 3 个实践项目，实施跨越整个学期的过程性考核，这部分占总成绩 90%。其次，课程依托

水杉码园平台，让学生自建 Git 代码仓库来锻炼实际工程能力，进而，结合学生提交的实际代码而非传统方式的单一报告来进行评价，完善了基于项目的工科生专业能力教学和评价方法。最后，课程采用项目作业“布置-提交-反馈”的教学模式，开展“即时反馈”的理论课和实践课教学，在学生提交作业后，及时批改并针对存在的问题在课堂上给予反馈和讲解，切实提高学生的实际动手能力并加深学生对理论知识的理解。

六、考核方式

考核方式：采用五级制，通过考勤、课堂表现、上机作业、实践项目和项目检查来考核。

成绩评定：采用过程性评估的方法，3 个实践项目的成绩占总成绩 60%，每个占 20%；5 个上机作业的成绩占总成绩 30%，每个占 6%；实践项目和上机作业会不定期组织作业检查；考勤和课堂表现组成平时成绩，占总成绩 10%。

评分规则：每个上机作业或实践项目的具体评级，根据实验要求的各项完成度判定：1) 不能正常跑出结果或没结果，根据情况给予不合格 (F)；2) 能正常跑出结果，可以给到及格 (C) 以上；3) 在 2) 的基础上，有完整过程描述，可以根据描述完整性和清晰度给到中 (B-) 或良 (B+)；4) 在 3) 的基础上，能对重难点给予适当分析，可以根据描述完整性和清晰度给到良 (B+) 或优 (A)；5) 完成选做附加题的，可以根据完成度酌情加分。

表-课程目标与考核方式对应关系

考核方式 课程目标	考勤	课堂表现	上机作业	实践项目
课程目标1	√	√		√
课程目标2	√	√	√	√
课程目标3	√	√	√	√
课程目标4	√	√	√	√
课程目标5	√	√		√

七、推荐教材和参考资料

1. MIT 公开课 Performance Engineering of Software Systems (编号 6.172)
2. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman. Compilers: Principles, Techniques, and Tools. Pearson Education, 2007.
3. John L. Hennessy, David A. Patterson. Computer Architecture: A Quantitative Approach (Sixth Edition). Morgan Kaufmann, 2019.
4. David J. Lilja. Measuring Computer Performance: A Practitioner's Guide. Cambridge University Press, 2004.
5. Denis Bakhvalov. Performance Analysis and Tuning on Modern CPUs. 2020.
6. Brendan Gregg. System Performance: Enterprise and the Cloud (Second Edition). Pearson Education, 2021.

八、评分标准

课程目标	评分标准				
	优 (A)	良 (B+)	中 (B-)	及格 (C)	不合格 (F)
课程目标 1	能完整理解和掌握软件系统性能优化的基本方法和原理。	能较为完整理解和掌握软件系统性能优化的基本方法和原理。	能理解和掌握部分软件系统性能优化的基本方法和原理。	大致了解软件系统性能优化的基本方法和原理。	不能有效理解软件系统性能优化的基本方法和原理。
课程目标 2	能完整掌握程序性能优化的基本原则与方法并编写高性能代码。	能较为完整掌握程序性能优化的基本原则与方法并编写高性能代码。	能掌握部分程序性能优化的基本原则与方法并编写高性能代码。	大致了解程序性能优化的基本原则与方法并编写高性能代码。	不能有效理解程序性能优化的基本原则与方法并编写高性能代码。
课程目标 3	能完整掌握性能数据采集、处理和分析的基本方法、技能和工具。	能较为完整掌握性能数据采集、处理和分析的基本方法、技能和工具。	能掌握部分性能数据采集、处理和分析的基本方法、技能和工具。	大致了解性能数据采集、处理和分析的基本方法、技能和工具。	不能有效掌握性能数据采集、处理和分析的基本方法、技能和工具。
课程目标 4	能完整掌握软件系统性能工程和基准测试的基本概念和方法并定位系统性能瓶颈。	能较为完整掌握软件系统性能工程和基准测试的基本概念和方法并定位系统性能瓶颈。	能掌握部分软件系统性能工程和基准测试的基本概念和方法并定位系统性能瓶颈。	大致了解软件系统性能工程和基准测试的基本概念和方法并定位系统性能瓶颈。	不能有效掌握软件系统性能工程和基准测试的基本概念和方法并定位系统性能瓶颈。
课程目标 5	能够完整理解软件系统性能优化的前沿研究内容。	能够较为完整理解软件系统性能优化的前沿研究内容。	能够理解部分软件系统性能优化的前沿研究内容。	大致了解软件系统性能优化的前沿研究内容。	不能有效理解软件系统性能优化的前沿研究内容。