

软件工程

董渊 2020

<http://soft.cs.tsinghua.edu.cn/blog/?q=se2020>

Department of Computer Science & Technology

介绍提纲

- 关于我们
- 课程介绍
- 教学目标
- 课程计划
- 教学方式
- 考察方式
- 课程要求
- Q&A

任课教师

- **董渊**，博士，清华大学计算机科学与技术系/研究生院
办公室：东主楼10区209，62794240
E-mail: dongyuan_AT_tsinghua.edu.cn
- **白晓颖**，博士，清华大学/军事科学院
- **吕勇强**，博士，清华大学信息技术研究院
E-mail: luyq_AT_tsinghua.edu.cn
- **李山山**，清华大学计算机科学与技术系
E-mail: lishanshan_AT_tsinghua.edu.cn
- **王瑀屏**，博士，清华大学计算机科学与技术系
E-mail: wyp_AT_tsinghua.edu.cn

助教团队

- **王蒙达，能源与动力工程系**
wangmd16_AT_mails.tsinghua.edu.cn
- **刘昊天，电机工程与应用电子技术系**
lht18_AT_mails.tsinghua.edu.cn
- **何彦妮，社会科学学院**
heyn19_AT_mails.tsinghua.edu.cn
- **郑和帆，土木工程系**
zhf20_AT_mails.tsinghua.edu.cn

系统支持

- 网络学堂（选课后的入口）+ “荷塘雨课堂”
访问地址：你懂的！
<https://tsinghua.yuketang.cn/pro/portal/home/>
- 课程blog，团队组建，课程作业展示(From2005)
访问地址：
<http://soft.cs.tsinghua.edu.cn/blog/?q=se2020>
- 代码托管，版本管理，代码提交、统计、测试和分析
访问地址：[github](#)/[gitlab](#)/[gitee](#) ...

软件工程

- 软件工程是计算机专业的一门重要课程，它对于培养良好的软件素质，提高**软件开发能力**与**软件项目管理能力**具有重要的意义。
- Standish Group Chaos Report
- | 年份 | 成功 | 有问题 | 失败 |
|------|-----|-----|-----|
| 1995 | 16% | 53% | 31% |
| 2009 | 32% | 44% | 24% |
| 2015 | 29% | 52% | 19% |

软件工程

- 因此，软件的开发是一个很困难的活动，要求很仔细地计划和执行。
- 软件工程就是在这样的背景下，由许多计算机科学家经过多方面的探索和总结而成形的。特别地，从多次失败中总结出来的经验和教训。
参看：《软件开发的滑铁卢》
- 目前依然处于初级阶段，发展十分迅速，不同的参与者（开发人员、管理人员）有不同的视角和观点。

教师建议

- 有刻苦学习，持之以恒的精神。本课程的性质与其说是一门技术性很强的课程，不如说更像一门**管理学科**的课程，很多规定都是国内外大师的经验总结，不能误认为无用或现在用不上而放弃学习。
- 能够学以致用，善于将理论与实际结合，在实践中认真按照软件工程要求做，才能巩固课内学到的知识，并能融会贯通。
- 重视实践，熟悉和使用主要的软件工程工具软件，锻炼能力。

教师建议

- 在课程中介绍了不少软件工程的基本原则和基本方法，在学习时，可以根据你的实践，认真思考一下为什么。**不但知其然，而且知其所以然**，这样会记得牢一些，不会看过就忘。
- 不少同学反映软件工程课程内容繁杂，不容易记忆。可以以**软件生存周期为主线**，对比各种开发方法的特点，找出它们的内在联系。对于各种管理方法，比较它们的作用，理出线索，可以帮助你学习记忆。
- 积极参与课堂讨论：说出你的困惑，寻求帮助；分享你的收获，回馈大家。
- **实践出真知**

上课时，同学们是这样滴 (20191227) ...





上课时，同学们还有这样滴(20180105)...



深夜下课时，同学们是这样滴(20171110)...



同学体会

“在这门课上，**收获也是与它的较高的难度相称的**。最基本的收获是了解和实践了软件开发的整个流程。教学与实践相结合总是能取得最好的效果，通过我们自己的尝试，我们对于软件开发的基本方法和流程、难点等有了一个不算浅薄的了解。前几节课上老师展示的软件开发极低的成功率背后的原因对于我们来说有了一个具象的体现。除此以外，这一份经历，无论最终结果如何，对于我们未来在进行代码乃至其它方面的合作都有着很大的意义。7~9人的团队对于一门课程的project来说已经可以算是很大的团队，因此我们就**面临了前所未有的沟通和协调的难题**。课上提供的小组讨论、管理工具能够对此提供一部分帮助，但成员间的主动的交流和互动是无法替代的。”

——（2019秋选课同学）

同学体会

“尽管已经上了两年多的计辅了，但是我的编程能力一直不是很好。在上软件工程的第一节课，听到这门课的要求的时候，我的**内心其实是崩溃的**。然而已经大四的我，也只能硬着头皮把这门课继续上下去。

一学期下来，我对于这门课**到底有多硬**的确深有体会，因为不管是从课程难度、知识量还是工作量来说，这门课的**要求都是极高的**。而线上线下结合的教学模式不（但）让学生花了更多的时间，也对我的自学能力提出了挑战。所幸在各位队友大佬们的帮助之下，我还是**成功地完成了**这门课程。在软件工程课上，虽然我们的大作业主要是要做一个软件，但是课程内容更多的是偏向对于一个软件工程的项目应该如何管理，如何落实各个项目，如何把控项目的各个环节，如何保证项目质量。软件工程的课程，老师通过一个大作业，让我们充分锻炼了项目管理和程序编码的能力，实现了编码与管理两开花，可谓是用心良苦。”

——（2018秋选课同学）

同学体会

“软件工程是一门信息量很大的课。我的感受是上这门课之前和之后自己的编程思想有了很大的改变。...我来自XX系，平时学习和科研深度**依赖编程**，而我总是在编程过程中纠结于一些小问题而失去总体的节奏。这门课教会我以自上而下的视角去建构一个软件（程序），并认识到这是一个**创造知识**的过程，从而避免轻视和错估这一过程。”

——（2017秋选课同学）

“一个学期的软件工程课结束了，但在这门课上的经历不会像某些课一样很快被我淡忘，反而将会一直留在我记忆中。其实在上这门课之前我是做好了**必死的准备**的，同班同学连续两年退课的事实告诉我这门课‘**水很深**’。但是早已经过数据结构洗礼的我怎能退缩，便婉拒了这次已是**第三次退课**的那个同学第四年陪他再战的请求，毅然地坚持走下去。”

——（2016秋选课同学）

参考教材

- 郑人杰 等，软件工程概论，2016，机械工业出版社
- “Software Engineering”
(7th Edition), Ian Sommerville
英文版，机械工业出版社，2004，7-111-15309
<http://www.comp.lancs.ac.uk/computing/resources/IanS/>
- “Software Engineering: A Practitioner’ s Approach”
(5th Edition), Roger S. Pressman,
McGraw-Hill, 2001，ISBN 0072496681
影印版，清华大学出版社，2001，ISBN 7-302-0413903
- “Software Engineering: Theory and Practice”
(2nd Edition), Shari Lawrence Pfleeger, 1998
Prentice Hall, ISBN 0130290491
影印版，高等教育出版社，2001，ISBN 7-04-010099-1
- “软件工程技术概论”
朱三元，钱乐秋，宿为民, 2002
科学出版社，ISBN 7-03-009940-0

教学目标

- 理解软件工程的基本概念，了解软件工程生命周期过程模型
- 熟悉软件工程的一般流程和主要活动，熟悉面向对象的软件分析、设计、建模的一般方法
- 掌握
 - 软件设计思想
 - 软件建模语言（UML）
 - 版本控制方法与工具
 - 基本的软件测试方法
- 培养
 - 软件开发
 - 团队合作
 - 展示自己
- 初步具备
 - 软件工程知识自主学习的能力
 - 将软件工程方法、技术应用于实践的能力

教学安排

- 软件工程概论
- 程序编码
- 软件过程
- 软件项目管理

- 软件需求
- 软件设计
- 软件测试

- 软件维护
- 软件重构
- 软件过程改进

教学方式

- **Education --> Learning** （联合国教科文组织）
 - 有组织有目的的知识传授活动
 - 能够导致学习的交流活动
- **多种方式、有效交流**
 - 课堂讲解
 - 课堂讨论
 - 作业讲评
- **“一、二、一”**
 - 一堆实例
 - 二次课堂讨论
 - 一个大作业

“一” 一些实例

- 历年“软件工程”课程的大作业
- 目标：软件工程方法、工具、活动展示
- 基本情况
 - 基本完整的文档
 - 相关代码
 - 可执行程序
- 贯穿整个课程、可在线访问
- 开放源码，可参与开发

“二” 二次课堂讨论

- 软件过程、需求与软件设计、项目进展
- 目标：答疑解惑、了解需求
- 基本情况
 - 互动方式
 - 师生问答
 - 项目进度讨论
 - 软件工程相关话题讨论
 - 作业讲评

“一” 一个大作业——Project

- 自己动手、亲身实践
- 目标：应用、巩固软件工程方法、工具
- 基本情况
 - 类型可选：开发项目、维护项目、项目分析
 - 队伍可选：7~9人，自组织，同一小组内来自同一院系的同学不超过3人
 - 内容可选：
根据自己的爱好、兴趣、方向和目标
 - 考察方式：演示、文档、讲解

“一” 一个大作业——Project

- 开发项目

- 个人或者团队选题并完成，
 - 鼓励选择与其他课程、研究课题相关的题目
- 开发语言、开发环境可选
- 符合软件工程的流程
 - 提交规范的需求分析、设计文档、测试报告和代码
- 报告和演示

- 实例

- Bombman
- ETeam : MouseTracker (2008)

“一” 一个大作业——Project

- 维护项目

- 个人或者团队选题并新增或改进部分功能
 - 鼓励选择与其他课程、研究课题相关的题目
 - 鼓励参加开源软件项目
- 开发语言、开发环境可选
- 符合软件工程的流程
 - 提交规范的需求分析、设计文档、测试报告和代码
- 报告和演示

- 实例

- 实用数据处理工具

关于大作业

来源：

- 1) 同学自己设计，或者其他课程的相关任务
- 2) 参加自由软件开发，完成一个新的功能或者特性
- 3) 教师结合课程情况，专门设计任务

大致安排：

- 由同学自行提出，并经过和教师讨论确定。
- 选课后酝酿，组队并讨论选题，课上汇报需求、设计，最后两周项目汇报（**110**，**100**加权）
- 要求分组独立完成并通过测试用例，提交源代码和相关文档，并能够演示结果，做10~15分钟的简要报告。采用GPL发布。

关键：

- 和课程相关，能够给同学以锻炼，工作量适中
- 具有较好的可操作性
- 鼓励和其它课程交叉

考察内容和评分标准

- 参与成绩 (40%)
 - 出勤情况、课堂讨论、社区参与
- 一次小作业 (10%)
 - 软件过程和项目建议 (小组成绩)
- 一个大作业 (40%)
 - 可自选题目，和老师、助教讨论决定 (小组成绩)
 - 等级制评分、严格限制A级小组比例 (不高于25%)
 - 每小组可推选1位MVP，成绩为小组成绩*1.1
- 一次个人总结 (10%)
 - 对软件工程的感受，对课程的建议和意见

课程要求

- **独立完成**
 - 要求独立完成任务一定独立完成
- **合作与分工**
 - 小组共同完成大作业时，明确每个人的分工
- **严禁抄袭**
 - 每个大作业题目和软件、文档在网上公开
 - 一旦发现，成绩归零
- **按时提交**
 - 延时24小时之内最高成绩80%
 - 延时24小时 ~ 7×24小时最高成绩60%
 - 延时1周以上者成绩为零