



软件开发环境国家重点实验室  
State Key Laboratory of Software Development Environment

# 软件工程基础

## 概论

欧阳元新 王德庆

2020年2月

- **欧阳元新，北京航空航天大学计算机学院(2005-)**
  - 先进计算机应用技术教育部工程研究中心，2005-
  - 博士，北京航空航天大学计算机学院，2005
  - 学士，北京航空航天大学计算机系，1997
- **办公室：新主楼 G950**
- **Email: oyyx@buaa.edu.cn**
- **答疑时间：通过邮件提前约定**

- **王德庆，北京航空航天大学计算机学院(2015-)**
  - 软件开发环境国家重点实验室，2015-
  - 硕、博士，北京航空航天大学计算机学院，2008、2012
  - 学士，山东大学计算机学院 2005
- **办公室：新主楼 G606**
- **Email: dqwang@buaa.edu.cn**
- **答疑时间：通过邮件提前约定**

**准备一张纸**

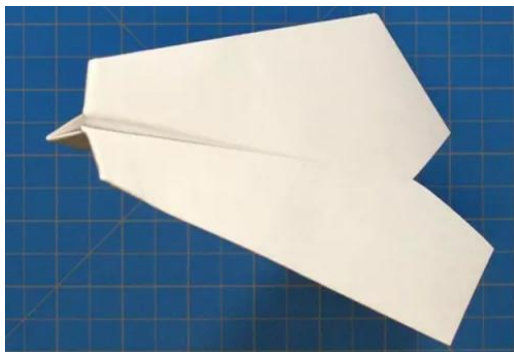
# 纸飞机



# 纸飞机——北航专业版

- 北航11种纸飞机折法，学会3种就成高手！

通过调整升降舵可使其  
螺旋飞行

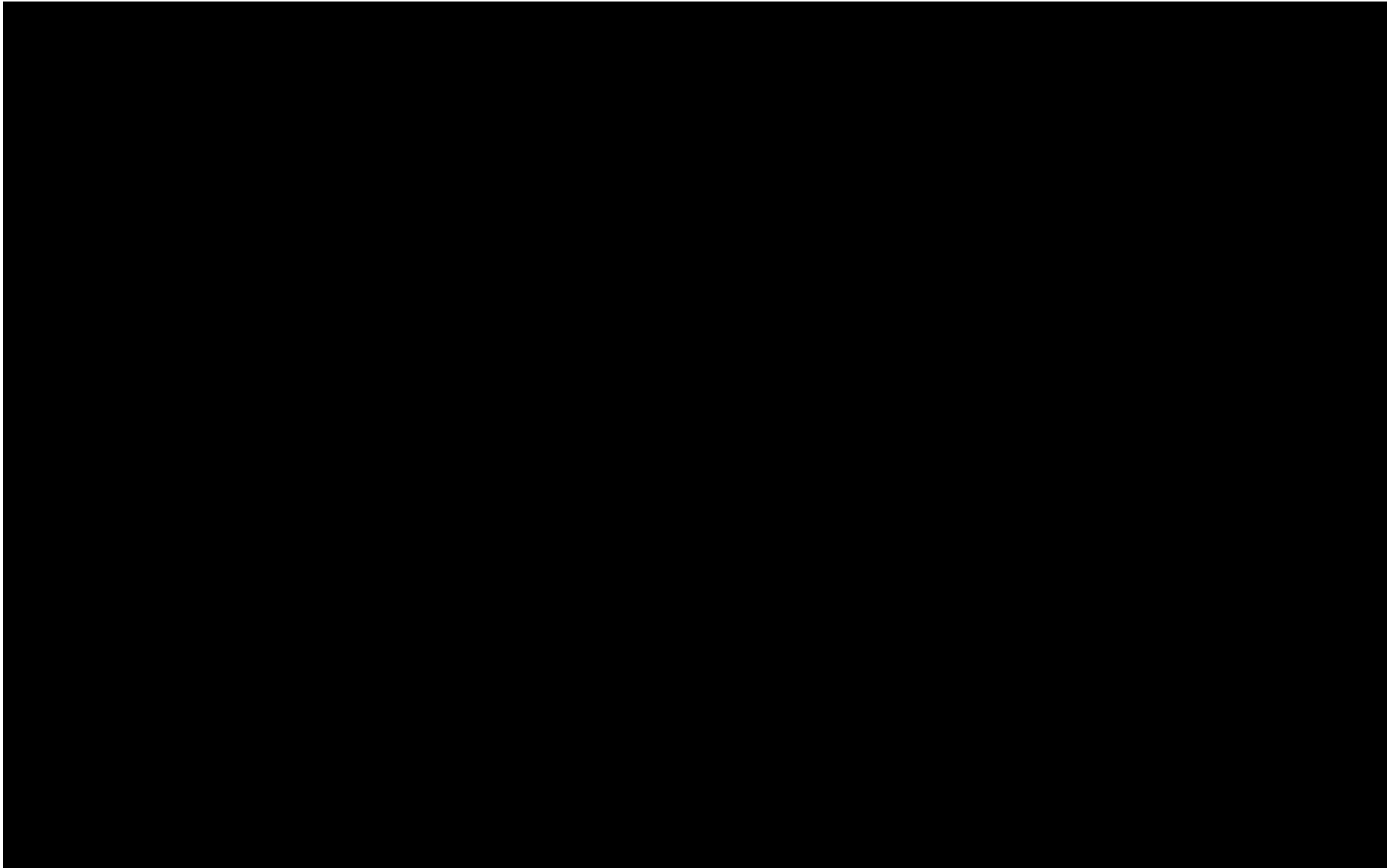


类似飞镖一样越飞越远    超小的体积，超长的飞行距离



<https://mp.weixin.qq.com/s/fX4DUkrglR8zf78huvCX4A>

# 纸飞机——世界级玩法



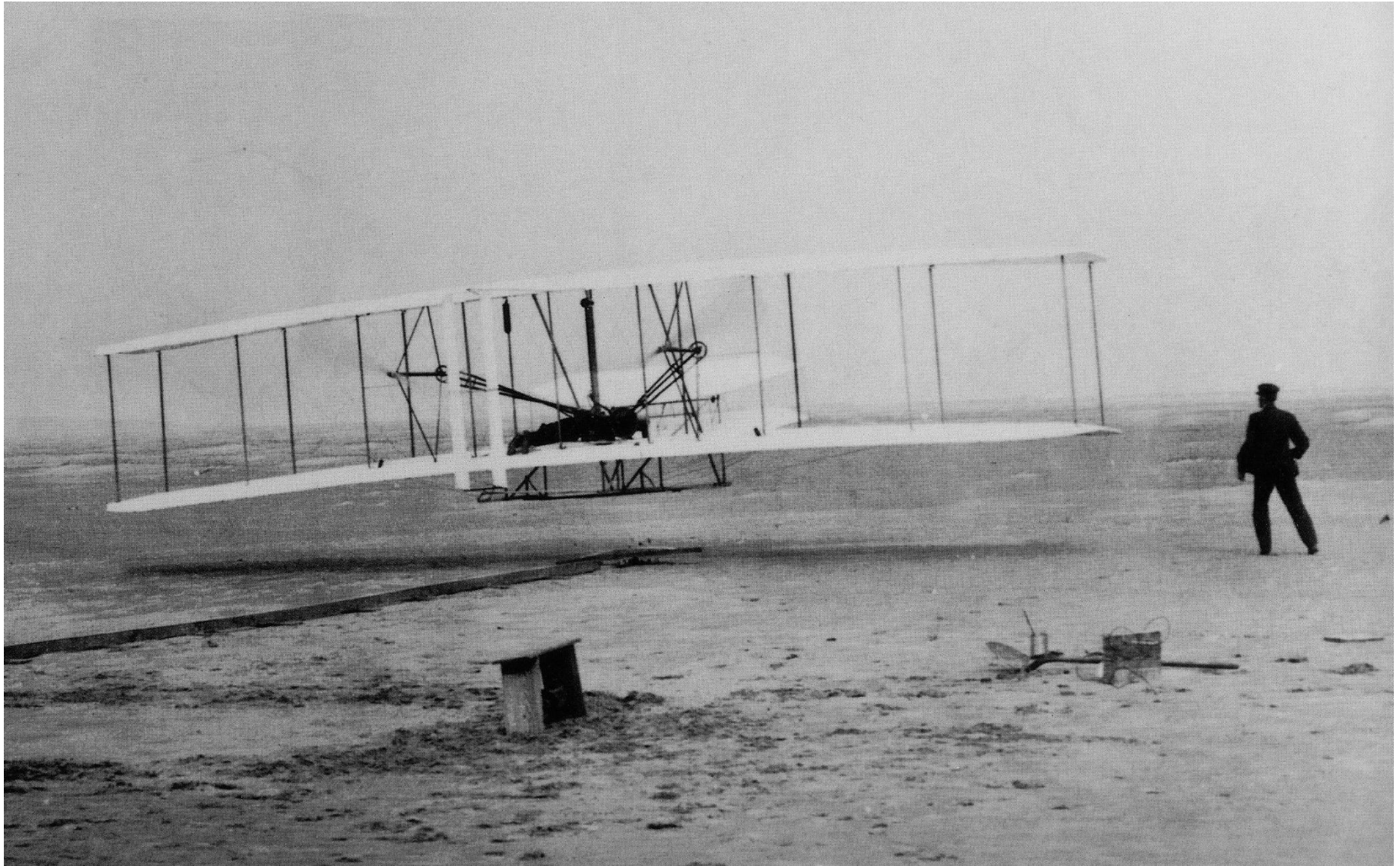


爱好者：草坪躺椅、气球





# 原型：Wright Brothers







# 航空业 vs. 软件业



## 软件开发

## 飞机

### 儿童玩具

- 用新的语言写一个 “hello world” 程序
- 时间：一个下午



### 爱好者

- 在Perl、Python、PHP、Ruby等中尝试一些新和酷的东西
- 时间：几天



### 原型

- 编写一个网站、工具、APP
- 时间：几周或者几个月



### 专业人士

- 通过编写软件谋生，比如Bill Gates
- 时间：几年，几百万甚至几千万美元的规模



**工程是将理论和知识应用于实践的科学，以便经济有效地解决问题。**

- 大规模的设计与建造
- 复杂问题与目标分解
- 团队协作与过程控制



软件工程 (Software Engineering) 是系统化、规范化、量化的方法在软件开发、操作和维护中的应用

**软件工程的目标 —— 创造 “足够好” 的软件**

- **规范:**

- 软件需求、设计、构建、测试和维护

- **较低的开发成本**

- 按时完成开发任务并及时交付
  - 实现客户要求的功能
  - 具有良好性能、可靠性、可扩展性、可移植性等
  - 软件维护费用低

**软件工程 (Software Engineering) 是系统化、规范化、量化的方法在软件开发、操作和维护中的应用**

**软件工程的目标 —— 创造 “足够好” 的软件**

- **相关领域：**

**计算机科学、计算机工程、管理学、数学、项目管理、质量管理、软件人类工程学（可用性）、系统工程、工业设计、用户界面设计**





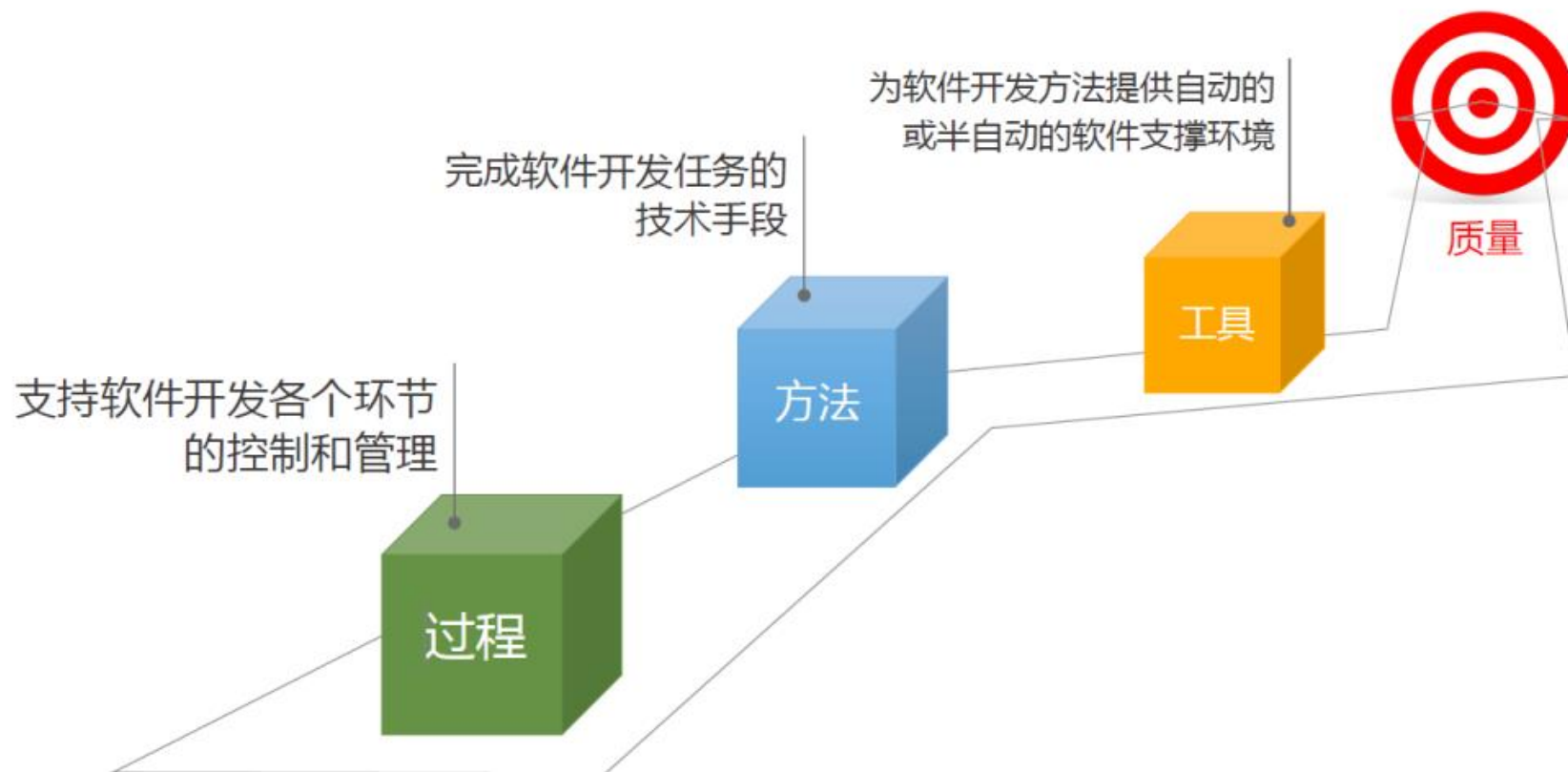
1. **Software Requirements**
2. **Software Design**
3. **Software Construction**
4. **Software Testing**
5. **Software Maintenance**
6. **Software Configuration Management**
7. **Software Engineering Management**
8. **Software Engineering Process**
9. **Software Engineering Models and Methods**
10. **Software Quality**
11. **Software Engineering Professional Practice**
12. **Software Engineering Economics**
13. **Computing Foundations**
14. **Mathematical Foundations**
15. **Engineering Foundations**

- **“Software engineering, of course, presents itself as another worthy cause, but that is eyewash: if you carefully read its literature and analyse what its devotees actually do, you will discover that software engineering has accepted as its charter ‘How to program if you cannot.’ ”**

**——Dijkstra**

- **软件工程是不是教那些不怎么会写程序的人开发软件？你怎么看？**

# 软件工程的要素



- **软件工程涉及以提高软件系统的有效性、可靠性和可维护性为目标的由软件开发流程约束的一系列技术和过程**

- 问题分析
- 需求分析
- 软件设计
- 软件实现
- 软件测试（单元测试、集成测试、性能测试、安全性测试）
- 软件维护（版本更新、Bug修订）
- 过程管理（人员管理、进度管理、.....）
- 质量管理
- 开发模式（个人软件过程、敏捷开发流程、瀑布开发模式、.....）
- .....



## 软件开发活动

问题定义

需求开发

软件设计

软件构造

软件测试

- 构想文档
- 用户故事

- 分析模型
- 软件需求规格说明

- 设计模型
- 软件体系结构文档
- 软件详细设计文档

- 源程序
- 目标代码
- 可执行构件

- 测试规程
- 测试用例
- 测试报告

## 软件开发管理

( 软件项目管理计划、软件配置管理计划、软件质量保证计划、评审记录..... )

# 为什么需要软件工程

- 开发复杂软件系统的需要

- 百万行以上的代码

- 软件是重要的社会基础设施，其质量关系着

- 巨额的经济利益
  - 人们的生命
  - 其上的商业运作



Windows XP大约  
40,000,000行代码





# 程序 vs. 服务

- 某数学老师要每周给同学出300道四则运算练习题。她想请你帮忙写一个程序，这样她每周都可以方便地出题并打印。
- 这个程序有很多种实现方式：
  - C/C++
  - C#/VB.net
  - Excel
  - Unix Shell
  - Emacs
  - Powershell/Vbscript
  - Perl
  - Python
- 现在估计写这个程序需要的时间

- **这个老师发现你的程序很好用，她的同事们都想要一份。这时你要考虑到下列新的要求：**
  - ❑ **题目避免重复**
  - ❑ **可扩展性**
  - ❑ **可定制（数量/打印方式）**
  - ❑ **具体定制**
    - 是否有乘除法
    - 是否有括号
    - 数值范围
    - 加减有无负数
    - 除法有无余数
    - 是否支持分数
    - 打印中每行的间隔可调整
- **现在估计做好这个软件服务需要多长时间。**

- 然后你发现全国的老师都有这个需求
- 甚至有好多家长都想要这个
- 那你要写一个网站满足大家的需求
- 现在估计完成这一软件服务需要的时间

- **程序 = 算法 + 数据结构**
- **软件 = 程序 + 数据 + 文档**
  - 程序：计算机可以接受的一系列指令，运行时可以提供所要求的功能和性能。
  - 数据：使得程序能够适当地操作信息的数据结构。
  - 文档：描述程序的研制过程、方法和使用的图文资料。
- **软件企业 = 软件 + 商业模式**
- **这门课程中合适的软件是什么？**
  - 规模万行以上
  - 满足特定用户群体需求

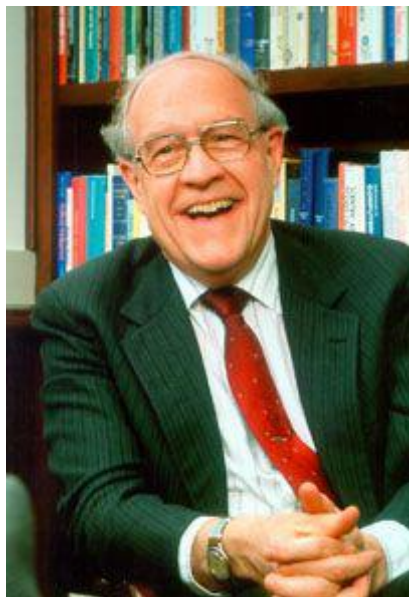
# 软件的本质特征



制造一架飞机和编写一个Word软件哪个的难度更大？

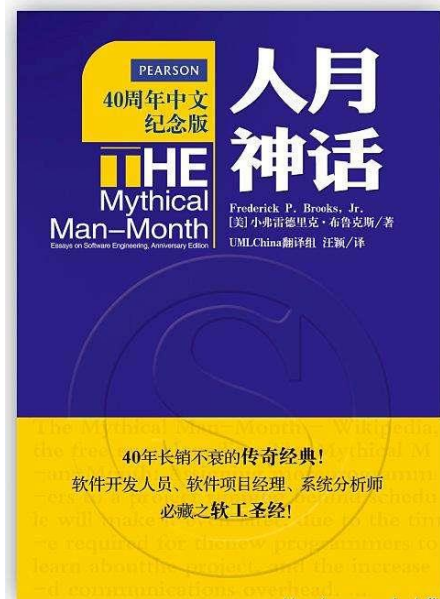
- 一架客机由数百万个单独的部件组成，需要上千人组装，但通常都能够按时按预算交付使用。
- 微软于1989年11月发布的Word最初版本，花费了55人年，大约有249,000行源代码，却晚了4年交付使用。

# 软件的本质特征



## No Silver Bullet 没有银弹

思考软件工程中的根本和次要问题  
Fredrick P. Brooks  
1986



- Fred Brooks是北卡罗莱纳大学的计算机科学教授，曾担任IBM OS360系统的项目经理，在计算机体系结构、操作系统和软件工程方面做出了里程碑式的贡献，于1999年荣获计算机领域最具声誉的图灵奖。



## • 软件分类

- 系统软件、应用软件、恶意软件、...

## • 软件特征

- 复杂性 (Complexity)
- 不可见性 (Invisible)
- 易变性 (Changeable)
- 服从性 (Conformity)
- 非连续性 (Discontinuity)

No Silver Bullet  
--Fred Brooks

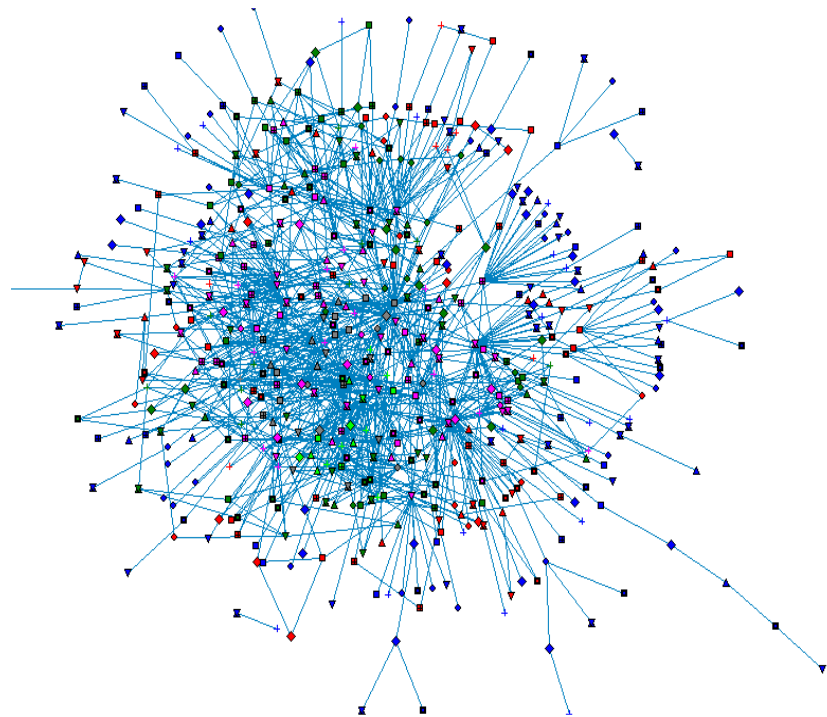
Software Engineering: The  
Current Practice  
--Vaclav Rajlich

# 软件的本质——复杂性

- Linux内核有630个函数，存在1814个函数调用。

注：图中将函数表示为节点，调用关系表示为边。

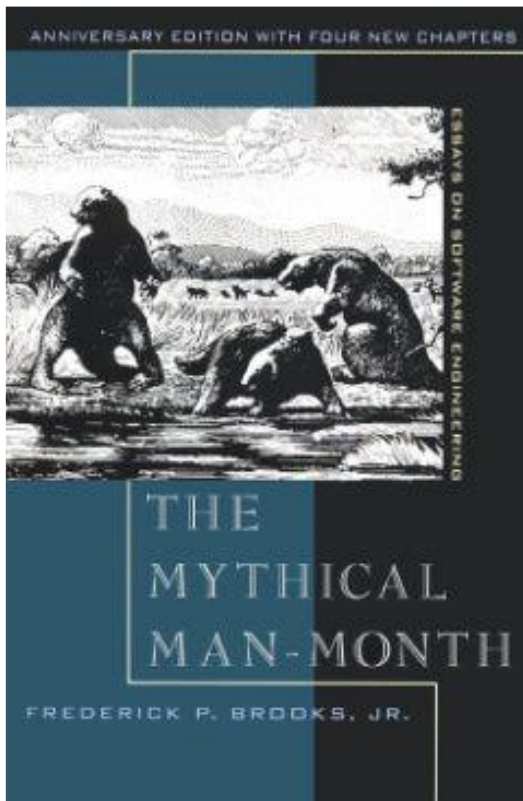
- Windows Vista: 从2001年开始研发，历时5年半，先后有9000位开发人员投入其中，耗资60亿美元，代码规模超过5000万行。由于整个系统过于庞杂，其开发管理相当混乱，以致于很多时间用在互相沟通和重新决定上。从Vista Beta 1进入公开测试以来，程序错误总数已经超过2万个



Windows Vista™

# 软件的本质——不可见性

- 软件是一种“看不见、摸不着”的逻辑实体，不具有空间的形体特征
- 开发人员可以直接看到程序代码，但是源代码并不是软件本身
- 软件是以机器代码的形式运行，但是开发人员无法看到源代码是如何执行的



软件人员太像“皇帝的新衣”故事中的裁缝了。当我来检查软件开发工作时，所得到的回答好象对我说：我们正忙于编织这件带有魔法的织物。只要等一会儿，你就会看到这件织物是及其美丽的。但是我什么也看不到，什么也摸不到，也说不出任何一个有关的数字，没有任何办法得到一些信息说明事情确实进行得非常顺利，而且我已经知道许多人最终已经编织了一大堆昂贵的废物而离去，还有不少人最终什么也没有做出来。

# 软件的本质——易变性



# FAQ 1



Question	Answer
What is software?	Computer programs and associated documentation. Software products may be developed for a particular customer or may be developed for a general market.
What are the attributes of good software?	Good software should deliver the required functionality and performance to the user and should be maintainable, dependable and usable.
What is software engineering?	Software engineering is an engineering discipline that is concerned with all aspects of software production.
What are the fundamental software engineering activities?	Software specification, software development, software validation and software evolution.
What is the difference between software engineering and computer science?	Computer science focuses on theory and fundamentals; software engineering is concerned with the practicalities of developing and delivering useful software.
What is the difference between software engineering and system engineering?	System engineering is concerned with all aspects of computer-based systems development including hardware, software and process engineering. Software engineering is part of this more general process.



# FAQ 2



Question	Answer
What are the key challenges facing software engineering?	Coping with increasing diversity, demands for reduced delivery times and developing trustworthy software.
What are the costs of software engineering?	Roughly 60% of software costs are development costs, 40% are testing costs. For custom software, evolution costs often exceed development costs.
What are the best software engineering techniques and methods?	While all software projects have to be professionally managed and developed, different techniques are appropriate for different types of system. For example, games should always be developed using a series of prototypes whereas safety critical control systems require a complete and analyzable specification to be developed. You can't, therefore, say that one method is better than another.
What differences has the web made to software engineering?	The web has led to the availability of software services and the possibility of developing highly distributed service-based systems. Web-based systems development has led to important advances in programming languages and software reuse.



# 如果某些地方出了差错.....

- **儿童玩具**

- 买一个新的

- **爱好者**

- 享受到了乐趣，去做其他事情

- **原型**

- 吸取教训，寻找其他途径并继续坚持

- **专业人士**

- 人们的生命、财产安全

## 小测验：

**如果在某个商业产品中，一个功能只有百万分之一的使用概率，那么你是否希望实现它并且每次客户使用你的产品时不厌其烦地教他们如何使用？**

- A. 根本就不打算做它**
- B. 如果我们没有时间就把它砍掉**
- C. 实现它，但是不需要告诉用户**
- D. 实现它，并且告诉用户**

## 安全须知

SAFETY INSTRUCTIONS

B737-700

为了您的安全 | For your safety

请仔细阅读本须知  
Please read instructions carefully  
听从机组的指令  
Follow crew's instructions  
如果您有其它问题，请与机组联系  
Please contact crews if you have further questions



飞行全程 | During the entire flight



禁止吸烟  
No smoking



禁止使用锂电池  
移动电源  
Do not use  
lithium battery  
mobile power



禁止使用主动发射无线电信号的便携式电子设备  
Do not use all portable transmitting devices



起飞与着陆 | Take-off and Landing

- 1 调直椅背  
Upright seatback
- 2 收起桌板  
Stow table
- 3 打开遮阳板  
Open window shade



系紧安全带  
Fasten seat belt  
tight & low

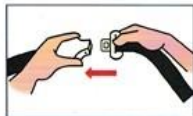


取下耳机  
No  
earphone



安全带 | Seat belt

系上  
Fasten



系紧  
Tighten



解开  
Unfasten



救生衣 | Life vest

位置  
Position



使用方法 Donning instructions



阅读后请放回原处  
Please return Instructions to seat pocket after reading.



## 安全须知

SAFETY INSTRUCTIONS

B737-700



氧气面罩 | Oxygen Mask



防冲击姿势 | Brace Position



应急出口 | Emergency Exit

应急灯 Emergency light



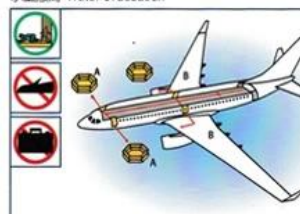
紧急出口A/B Emergency exit A&B



撤离 Ground evacuation



水上撤离 Water evacuation



HD-MU 05A







**这是否是一个有效的需求.....**

- **让一架飞机安全的在水面上着陆？**

# 飞机能否在河中着陆？



## • 创造 “足够好” 的软件

- ❑ 用户满意度
- ❑ 可靠性
- ❑ 软件过程质量
- ❑ 可维护性

Product characteristic	Description
Maintainability	Software should be written in such a way so that it can evolve to meet the changing needs of customers. This is a critical attribute because software change is an inevitable requirement of a changing business environment.
Dependability and security	Software dependability includes a range of characteristics including reliability, security and safety. Dependable software should not cause physical or economic damage in the event of system failure. Malicious users should not be able to access or damage the system.
Efficiency	Software should not make wasteful use of system resources such as memory and processor cycles. Efficiency therefore includes responsiveness, processing time, memory utilisation, etc.
Acceptability	Software must be acceptable to the type of users for which it is designed. This means that it must be understandable, usable and compatible with other systems that they use.



- **软件工程涉及以提高软件系统的可靠性和可维护性为目标的由软件开发流程约束的一系列技术和过程**
- **缺乏质量 = 缺陷 (bug)**

# 史上第一个计算机Bug——1947 Grace

9/9


0800 Antan started  
1000 " stopped - antan ✓

1300 (032) MP-MC  $\begin{cases} 1.2700 & 9.037847025 \\ 2.130476415 & 9.037846995 \text{ correct} \end{cases}$   
(033) PRO 2  $\begin{cases} 1.582647000 \\ 2.130476415 \end{cases}$   
convd  $\begin{cases} 2.130476415 \\ 2.130676415 \end{cases}$

Relays 6-2 in 033 failed special speed test  
in relay 10,000 test.

Relays changed

1100 Started Cosine Tape (Sine check)  
1525 Started Multi-Adder Test.

1545  Relay #70 Panel F  
(moth) in relay.

First actual case of bug being found.

1630 Antan started.  
1700 closed down.

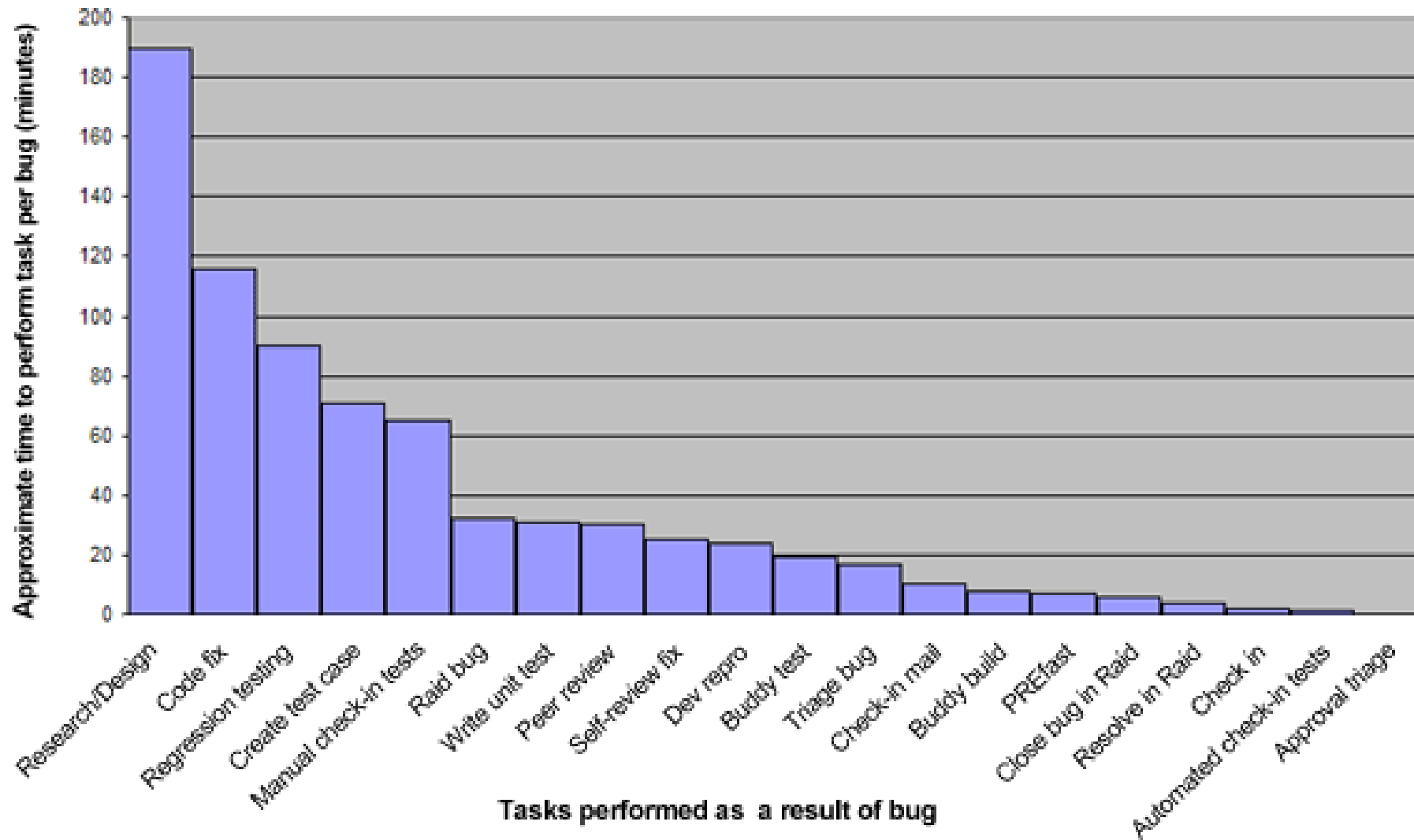
Relay 2145  
Relay 3371

- 在专业软件开发中，缺陷是代价高昂的
- 冲击波安全漏洞(Blaster)
  - 3行代码
  - 花费了微软10亿美元
- MSFT的研究表明
  - 修复一个缺陷平均需要花费12小时，大约900美元

# 冲击波的影响

企业的规模 /PC的数量	中等或重大影响 的比例	受感染企业的损失(硬件、软件和 生产力)
100K - 500K	55.0 percent	US\$2,400,000
50K - 95K	41.6 percent	US\$4,228,000
10K - 50K	33.8 percent	US\$3,458,741
5K-10K	39.0 percent	US\$1,452,288
1K - 5K	30.6 percent	US\$474,847
500 - 1000	19.8 percent	US\$84,207
100 - 500	11.1 percent	US\$79,863

Costs to Fix Bug





- 昂贵的代码复制，花费4亿美金的阿丽亚娜5型烟花



- 什么是bug?
- 与预期不符的软件行为
  - 崩溃、数据丢失、挂起
  - 没有实现宣称的功能
  - 使用困难
- “预期” 从哪里来
  - 从客户/用户中来



## 某大学食堂豆腐丝充肉荤引大学学生不满

### • 学生：

- ❑ 荤菜里肉丝寥寥，豆腐丝很多
- ❑ 食堂有愚弄学生之嫌
- ❑ It' s a bug!

### • 校方：

- ❑ 豆腐丝 = 素肉
- ❑ 加入素肉为营养均衡，“植物蛋白”
- ❑ It' s a feature!

- 小汽车

- QQ
  - Hyundai
  - BMW

- 新车出厂, 它们都质检合格
- 它们的质量有区别么?
- 但是它们找到了各自合适的顾客





图灵奖获得者Tony Hoare比较过计算机科学和软件工程的不同侧重点：

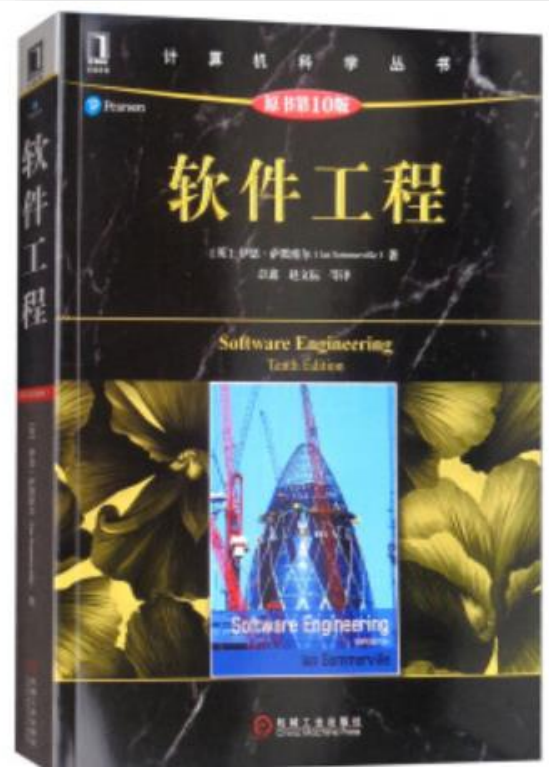
计算机科学	软件工程
发现和研究长期的、客观的真理	短期的实际结果（具体的软件会过时）
理想化的	对各种因素的折衷
确定性	对不确定性和风险的管理
完美	足够好
通用性	具体的应用
各个学科独立深入研究，做出成果	关注和应用各个相关学科的知识，解决问题
理论的统一	百花齐放的实践方法
强调原创性	最好的、成熟的实践方法
形式化，追求简明的公式	在实践中建立起来的灵感和直觉
正确性	可靠性





软件开发环境国家重点实验室  
State Key Laboratory of Software Development Environment

# 软件工程基础 课程简介



- 软件工程（原书第10版），Ian Sommerville著，彭鑫等译

- 构建之法 现代软件工程（第三版）邹欣 著

- 通过**团队协作**使用软件开发工具按照**软件工程方法**来开发**高质量**并且**可用的复杂软件系统**的能力
- 在这堂课上，你将
  - **体验软件的全生命周期**
    - 学习不同的角色以及他们之间的交互
    - 可以总结经验和教训
  - **开发一个你骄傲的软件/服务**
    - 可用的实际软件
  - **学习如何与其他人在一个团队里工作**
    - 在冲突中学会如何构建一个成功的团队
  - .....

## • 什么是软件工程

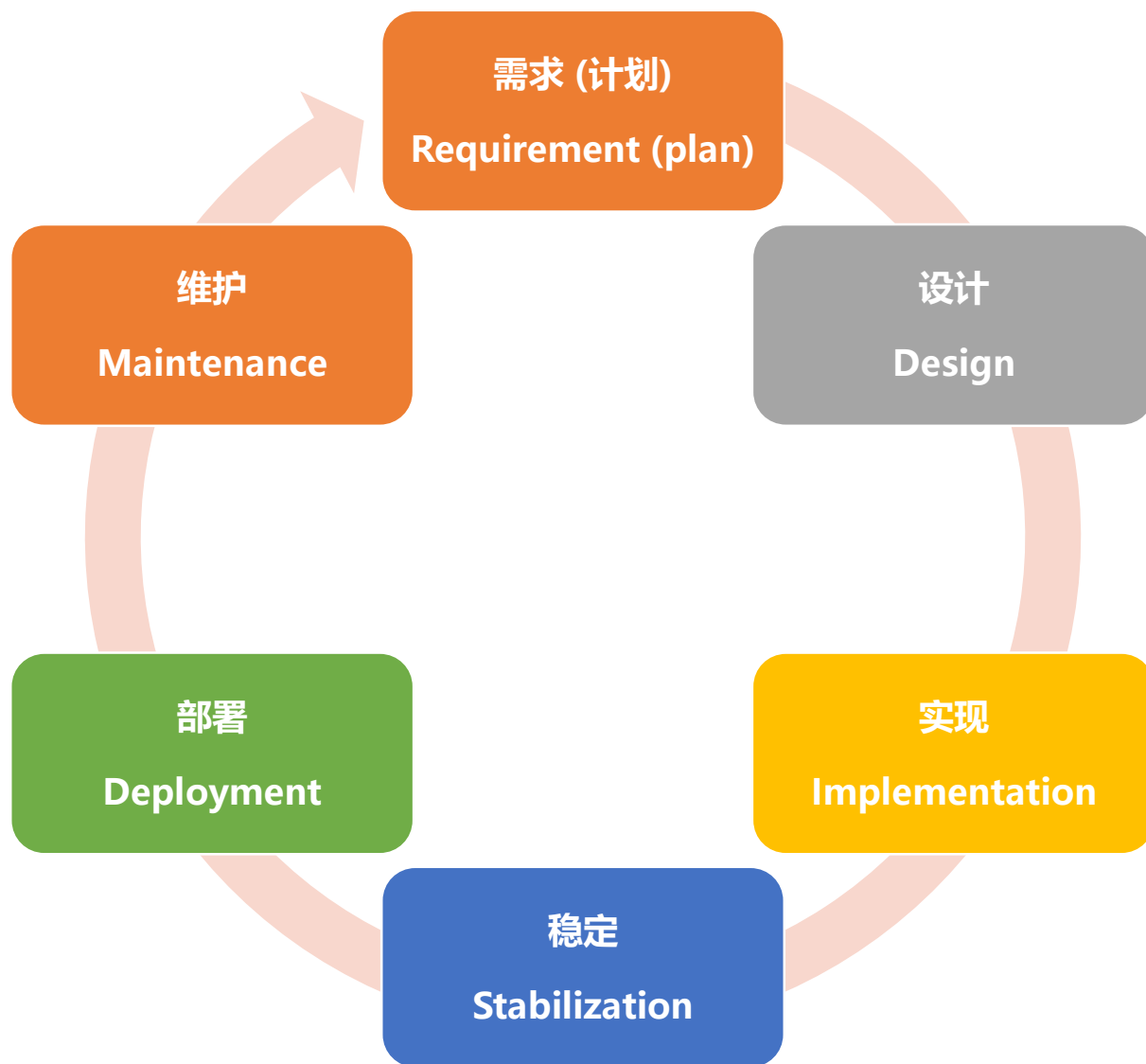
- 需求分析
- 规格说明书
- 开发模式
  - 瀑布模型
  - 增量式开发
  - 敏捷开发方法 (Agile)
- 软件生命周期各个阶段所要用的工具
  - 例如，单元测试、性能测试、Bug诊断、配置

## • 如何利用软件工程方法通过团队协作开发一个复杂系统？

- **讲授高满意度的课程内容**
  - 评价标准：好的反馈
- **展示怎样交付一个实际的软件项目**
  - 评价标准：成功的项目和/或经验教训
  - 项目的影响
- **从你们身上学习...**
  - 评价标准：you show me...



# 软件工程一般如何讲授



## • 以复杂软件开发为主线

- 课上讲授8次
- 评审、研讨与课上汇报 8次

## • 团队项目

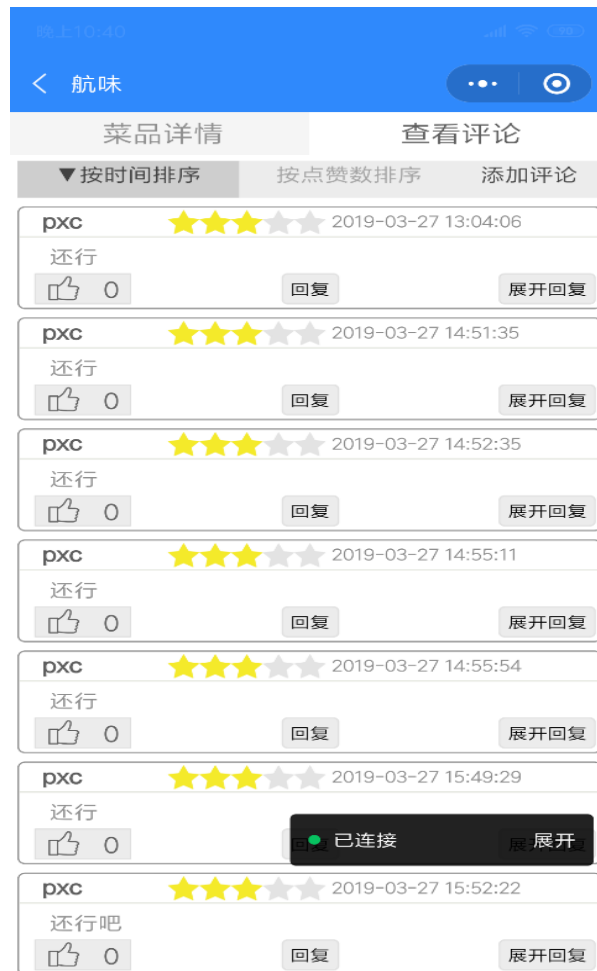
- 4人一组，个别5人小组需由教师确认后组建
- 角色分工与轮换：需求分析人员、开发人员、测试人员、项目经理等多种角色

- 根据我们从2012年到2019年的统计数据

- ❑ 团队项目的代码量一般在万行以上，优秀的项目一般达到了2万行左右
- ❑ 90%以上的同学能够积极地参与实践训练
- ❑ 团队项目中个人的平均代码量~1700行，优秀团队中个人平均代码量超过了2000行
- ❑ 学生平均每周工作时间~12小时
- ❑ 通过一个学期的学习，大部分学生们对自己能力的评价上从“少量的书面基本知识”提高到了“基本的理论和实践知识，可以通过企业的面试”；成绩优秀的同学对自己能力的评价达到“具有经实战考验过的技能，可通过最高水平企业的面试”

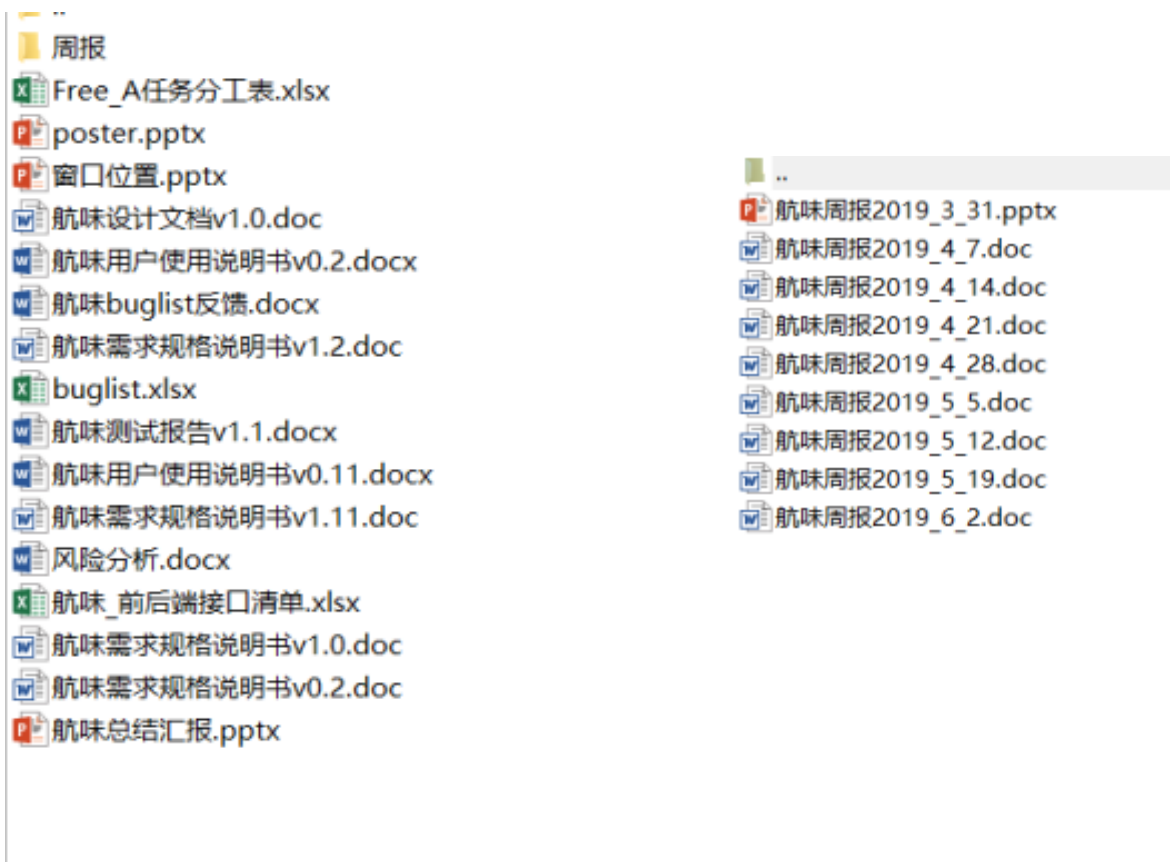
# 历届优秀作品介绍

## • 航味：一款面向北航师生的食堂饭菜评价系统



# 历届优秀作品介绍

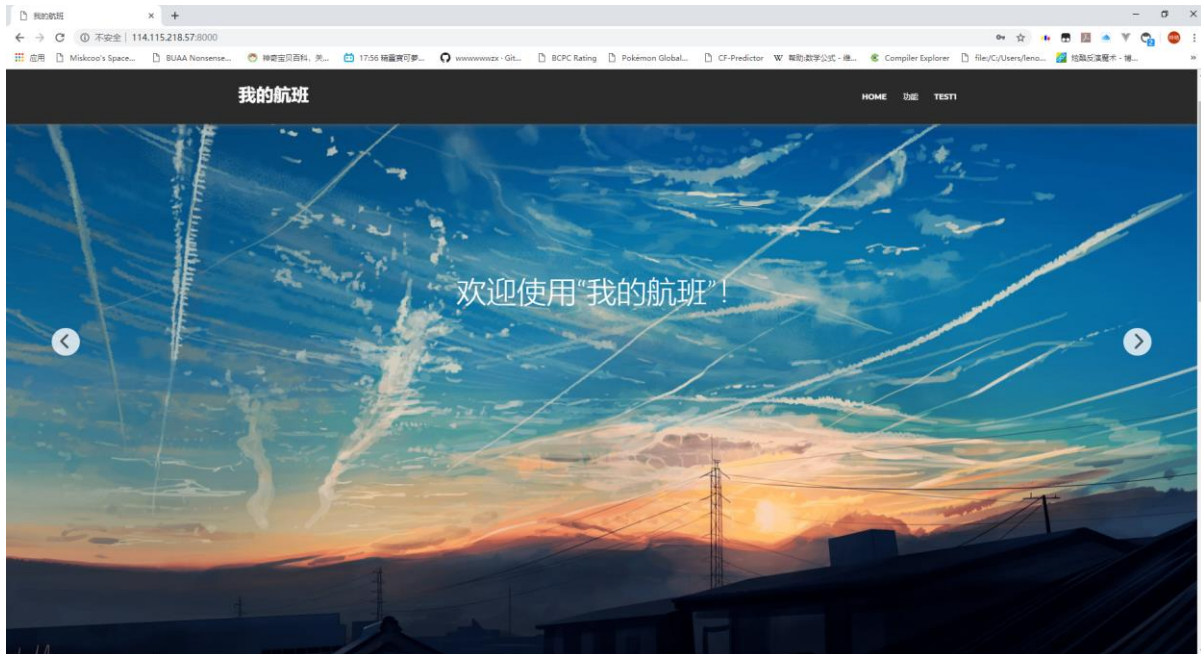
## • 航味：一款面向北航师生的食堂饭菜评价系统





# 历届优秀作品介绍

- **我的航班：从若干种信息源中获取航班、机场等实时信息**  
对信息加以处理和聚合，并充分利用获取到的信息，可用客户端获取相应的信息，或通过邮件、推送等方式获得相关信息




## • OmniSci: 项目发布与数据收集系统

成果库


讨论区

上传图片


这个项目由任厚兴大佬发起，旨在探索儿童心理的变化，为儿童的健康成长提供可靠的指导




Giggle 2019/04/01




Giggle 2019/04/01




Giggle 2019/04/01




Houxing 2019/04/01




Houxing 2019/04/01




Houxing 2019/04/01



Feng 2019/04/01



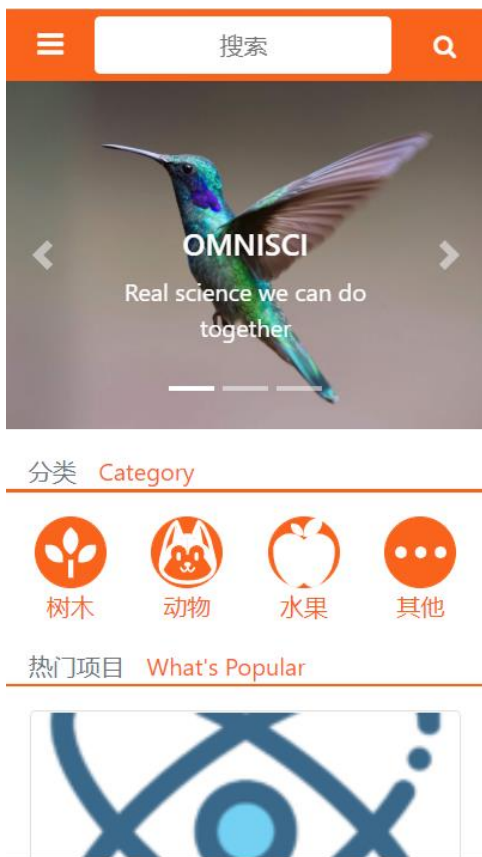
Feng 2019/04/01



Feng 2019/04/01


# 历届优秀作品介绍


## • OmniSci: 项目发布与数据收集系统





# 往届同学开发的项目

Phylab 实验 社区 工具 反馈

[Home](#) [Course](#) [Questions](#)

OR [register](#) [login](#)

### Software Engineering


Hojas del Verano

Design, Development and Maintenance


### javascript

### java


### c#




BUAA SOFTWARE ENGINEERING



$F=ma$



Project interface showing a search bar and a list of items.



### Crawler

DataHandler

[Start](#) [Stop](#)

### Overview

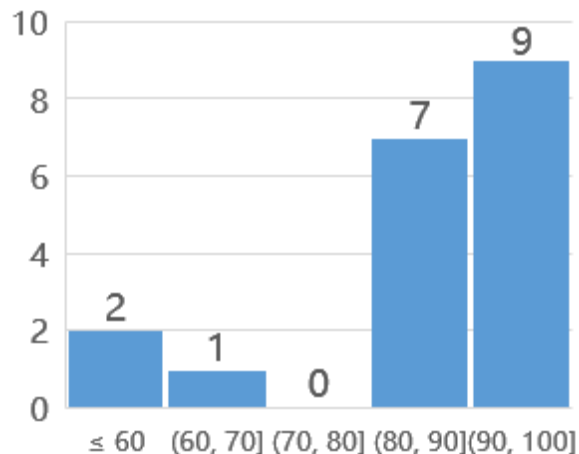
状态:	working
开始爬取时间:	2017-0-4-5-41-19
已爬取文件总数:	5800
当前启动的爬虫线程数:	1

### Details

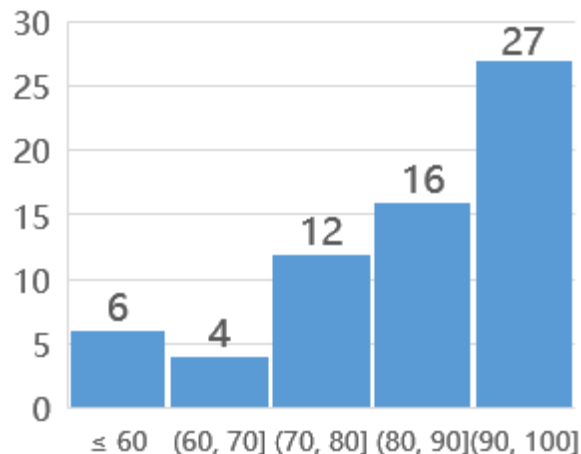
已爬取Html数量:	3348
已爬取Pdf数量:	0
已爬取Doc数量:	2
已爬取图片数量:	2450

# 往届分数分布情况

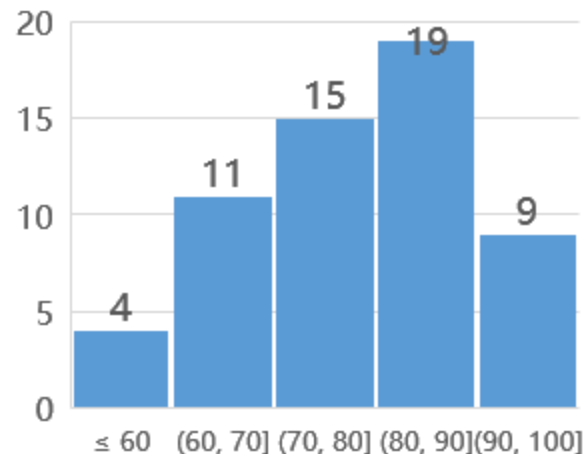
## 2016年软件工程基础成绩分布



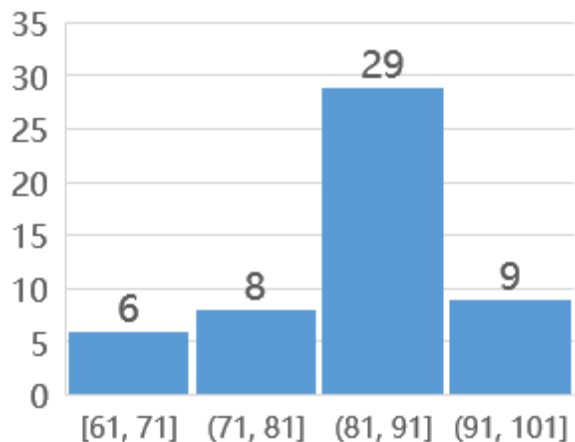
## 2015年软件工程基础成绩分布



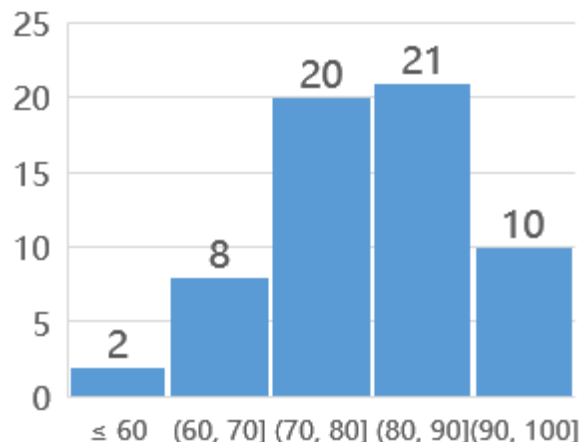
## 2014年软件工程基础成绩分布



## 2013年软件工程基础成绩分布



## 2012年软件工程基础成绩分布





- **软件工程这门课中，我不仅学到了技术，还学到了团队协作。**
- **后端的 API 设计离不开与 PM 讨论需求、与前端手机端商量接口、与组员商量分工**
- **同时我还学到了在多人协作中，做好代码的版本管理是多么重要，以及生产环境和开发环境的分离的重要性。**
- **我们因为在生产和开发环境中使用同一个数据库，经常把数据库表搞得乱七八糟，所幸没有重要数据。**

- DDL的重要性
- 里程碑的重要性
- 一个人可以跑得很快，一群人可以跑得更远
- 感谢组长组员们一路的帮助与陪伴
- 当代码不再是一个人的随心所欲
- 从入门到重构，好界面都是磨出来的
- 计划就是为了适应变化
- 接触完整的项目，体验开发者身份，学习协同与合作

陪伴我们一个学期的软工已经进入了尾声。在这长达十几周的长期奋战中，我由一个完全搞不懂html，css和js作用的前端小白，经过一番波折，成长为一个能稍微读懂，略微套用，修改一点前端UI样式的前端大白了。在学习中，我深刻体会到了，站在巨人肩膀上的优越性，以及不要重复造轮子的心得体会。最后，开源是个好文明，angularjs和bootstrap都是好东西。

另外，在此我要特别感谢组内的同学们给予我的帮助，如果没有他们，我一定无法完成这门艰巨的课程。也感谢老师和助教们的辛勤付出。谢谢大家。

- 使用你的软件的真实用户

- 展示用户数量

- 所有材料公开

- 每个团队公布可实施的进度表

- 使用真实数据和人对项目进行评判：

- 数据 (速度、用户数、用户反馈)
  - 评估小组 (其他同学、老师、专业软件工程师等)

- 真实的反馈

- 自我评价问卷调查
  - 事后分析
  - 对于课程的反馈

## • 积极主动

- ❑ 问问题，各种问题
- ❑ 告诉我你所想，我可能会按你说的去做
- ❑ 犯错误，犯很多的错误
- ❑ 给我直接的反馈 (课堂上、邮件等)

## • 实际动手操作

- ❑ 理论上，理论和实践是一样的
- ❑ 实践中，它们是不一样的

## • 做的事情需要有结果

- ❑ 让我们看到你所做工作的结果



## • 课程作业

- 个人作业
- 团队项目

## • 4次评审

- 根据4次评审得到团队的整体得分
- 团队成员协商确定每个成员的贡献度
- 不及格没有补考，需重修

## • 日常重要制度

- 周报：需要填写周报，并公开或课内做汇报

- 软件过程管理工具
  - 华为软开云 [统一分配账号]
- 代码管理工具
  - Github或华为软开云
- 软工班级微信群
- 工具使用说明
  - 助教群里提供

# Q&A