形式化方法导引

第 0 章 学前准备

黄文超

http://staff.ustc.edu.cn/~huangwc/fm.html

2022 年 春季学期

课前准备

助理教师: 解围

邮箱: xxieww@ustc.edu.cn

课程 QQ 群:

参考教材 (与阅读)

- Logic in Computer Science: Modelling and Reasoning about Systems. 2004
- 《形式化方法导论》. 2015.
- Formal Methods: An Appetizer . 2019
- Handbook of Model Checking. 2018
- Introduction to the Theory of Computation. 2018
  - 研究生课程：《形式语言与计算复杂性》
- 《致命 Bug: 软件缺陷的灾难与启示》. 2018

注：

- **不要求**阅读上述任何教材，学习资料会尽可能在 PPT 中展示
  - 每个 Slides 有两个版本：演示版 (PPT)、阅读版 (Article)(附加详细注释)
- 可选择感兴趣的方向进一步深入阅读

参考教材 (与阅读)

- Logic in Computer Science: Modelling and Reasoning about Systems. 2004
- 《形式化方法导论》. 2015.
- Formal Methods: An Appetizer . 2019
- Handbook of Model Checking. 2018
- Introduction to the Theory of Computation. 2018
  - 研究生课程：《形式语言与计算复杂性》
- 《致命 Bug: 软件缺陷的灾难与启示》. 2018

注：

- **不要求**阅读上述任何教材，学习资料会尽可能在 PPT 中展示
  - 每个 Slides 有两个版本：演示版 (PPT)、阅读版 (Article)(附加详细注释)
- 可选择感兴趣的方向进一步深入阅读

参考教材 (与阅读)

- Logic in Computer Science: Modelling and Reasoning about Systems. 2004
- 《形式化方法导论》. 2015.
- Formal Methods: An Appetizer . 2019
- Handbook of Model Checking. 2018
- Introduction to the Theory of Computation. 2018
  - 研究生课程：《形式语言与计算复杂性》
- 《致命 Bug: 软件缺陷的灾难与启示》. 2018

注：

- **不要求**阅读上述任何教材，学习资料会尽可能在 PPT 中展示
  - 每个 Slides 有两个版本：演示版 (PPT)、阅读版 (Article)(附加详细注释)
- 可选择感兴趣的方向进一步深入阅读

实验参考

- SAT/SMT by Example
- NuSMV
- Website: ProVerif
- Website: Event B
- Website: Coq Proof Assistant
- CCF A、B 类论文
  - 网络与信息安全
  - 软件工程/系统软件/程序设计语言

学习目标

- 【了解】【背景】形式化方法的**需求背景、当前现状及未来挑战**
- 【掌握】【理论】形式化方法的**基础理论及应用方法**
  - **通用**形式化方法相关术语
  - SAT、SMT、模型检测等核心求解算法
- 【初步掌握】【编程】各种形式化语言的使用
  - 形式化建模的方法?
  - 安全属性的建立?
  - 如何验证、证明?
  - 与传统编程的区别?
- 【了解】理解各工具的适用场景

学习目标

- 【了解】【背景】形式化方法的**需求背景、当前现状及未来挑战**
- 【掌握】【理论】形式化方法的**基础理论及应用方法**
  - 通用形式化方法相关术语
  - SAT、SMT、模型检测等核心求解算法
- 【初步掌握】【编程】各种形式化语言的使用
  - 形式化建模的方法?
  - 安全属性的建立?
  - 如何验证、证明?
  - 与传统编程的区别?
- 【了解】理解各工具的适用场景

学习目标

- 【了解】【背景】形式化方法的**需求背景、当前现状及未来挑战**
- 【掌握】【理论】形式化方法的**基础理论及应用方法**
  - **通用**形式化方法相关术语
  - SAT、SMT、模型检测等核心求解算法
- 【初步掌握】【编程】各种形式化语言的使用
  - 形式化建模的方法?
  - 安全属性的建立?
  - 如何验证、证明?
  - 与传统编程的区别?
- 【了解】理解各工具的适用场景

学习目标

- 【了解】【背景】形式化方法的**需求背景、当前现状及未来挑战**
- 【掌握】【理论】形式化方法的**基础理论及应用方法**
  - **通用**形式化方法相关术语
  - SAT、SMT、模型检测等核心求解算法
- 【初步掌握】【编程】各种形式化语言的使用
  - 形式化建模的方法?
  - 安全属性的建立?
  - 如何验证、证明?
  - 与传统编程的区别?
- 【了解】理解各工具的适用场景

学习目标

- 【了解】【背景】形式化方法的**需求背景、当前现状及未来挑战**
- 【掌握】【理论】形式化方法的**基础理论及应用方法**
  - **通用**形式化方法相关术语
  - SAT、SMT、模型检测等核心求解算法
- 【初步掌握】【编程】各种形式化语言的使用
  - 形式化建模的方法?
  - 安全属性的建立?
  - 如何验证、证明?
  - 与传统编程的区别?
- 【了解】理解各工具的适用场景

学习目标

- 【了解】【背景】形式化方法的**需求背景、当前现状及未来挑战**
- 【掌握】【理论】形式化方法的**基础理论及应用方法**
  - **通用**形式化方法相关术语
  - SAT、SMT、模型检测等核心求解算法
- 【初步掌握】【编程】各种形式化语言的使用
  - 形式化建模的方法?
  - 安全属性的建立?
  - 如何验证、证明?
  - 与传统编程的区别?
- 【了解】理解各工具的适用场景

学习目标

- 【了解】【背景】形式化方法的**需求背景、当前现状及未来挑战**
- 【掌握】【理论】形式化方法的**基础理论及应用方法**
  - **通用**形式化方法相关术语
  - SAT、SMT、模型检测等核心求解算法
- 【初步掌握】【编程】各种形式化语言的使用
  - 形式化建模的方法?
  - 安全属性的建立?
  - 如何验证、证明?
  - 与传统编程的区别?
- 【了解】理解各工具的适用场景

学习目标

- 【了解】【背景】形式化方法的**需求背景、当前现状及未来挑战**
- 【掌握】【理论】形式化方法的**基础理论及应用方法**
  - **通用**形式化方法相关术语
  - SAT、SMT、模型检测等核心求解算法
- 【初步掌握】【编程】各种形式化语言的使用
  - 形式化建模的方法?
  - 安全属性的建立?
  - 如何验证、证明?
  - 与传统编程的区别?
- 【了解】理解各工具的适用场景

学习水平评价

1. 【编程小作业】知道有这些东西，会编译这些工具
2. 【平时作业】能看懂别人的代码
3. 【平时作业、编程小作业】会修改别人的代码，能看懂验证结果
4. 【编程小作业】会修改别人的代码，能手工验证
5. 【编程大作业】会选择、并使用合适的工具进行建模、验证一个模型
6. 【编程大作业】【可选、加分】会改进工具，提升工具的效率
7. 【可选、满分】完全自己编写工具

学习水平评价

1. 【编程小作业】知道有这些东西，会编译这些工具
2. 【平时作业】能看懂别人的代码
3. 【平时作业、编程小作业】会修改别人的代码，能看懂验证结果
4. 【编程小作业】会修改别人的代码，能手工验证
5. 【编程大作业】会选择、并使用合适的工具进行建模、验证一个模型
6. 【编程大作业】【可选、加分】会改进工具，提升工具的效率
7. 【可选、满分】完全自己编写工具

学习水平评价

1. 【编程小作业】知道有这些东西，会编译这些工具
2. 【平时作业】能看懂别人的代码
3. 【平时作业、编程小作业】会修改别人的代码，能看懂验证结果
4. 【编程小作业】会修改别人的代码，能手工验证
5. 【编程大作业】会选择、并使用合适的工具进行建模、验证一个模型
6. 【编程大作业】【可选、加分】会改进工具，提升工具的效率
7. 【可选、满分】完全自己编写工具

学习水平评价

1. 【编程小作业】知道有这些东西，会编译这些工具
2. 【平时作业】能看懂别人的代码
3. 【平时作业、编程小作业】会修改别人的代码，能看懂验证结果
4. 【编程小作业】会修改别人的代码，能手工验证
5. 【编程大作业】会选择、并使用合适的工具进行建模、验证一个模型
6. 【编程大作业】【可选、加分】会改进工具，提升工具的效率
7. 【可选、满分】完全自己编写工具

学习水平评价

1. 【编程小作业】知道有这些东西，会编译这些工具
2. 【平时作业】能看懂别人的代码
3. 【平时作业、编程小作业】会修改别人的代码，能看懂验证结果
4. 【编程小作业】会修改别人的代码，能手工验证
5. 【编程大作业】会选择、并使用合适的工具进行建模、验证一个模型
6. 【编程大作业】【可选、加分】会改进工具，提升工具的效率
7. 【可选、满分】完全自己编写工具

学习水平评价

1. 【编程小作业】知道有这些东西，会编译这些工具
2. 【平时作业】能看懂别人的代码
3. 【平时作业、编程小作业】会修改别人的代码，能看懂验证结果
4. 【编程小作业】会修改别人的代码，能手工验证
5. 【编程大作业】会选择、并使用合适的工具进行建模、验证一个模型
6. 【编程大作业】【可选、加分】会改进工具，提升工具的效率
7. 【可选、满分】完全自己编写工具

学习水平评价

1. 【编程小作业】知道有这些东西，会编译这些工具
2. 【平时作业】能看懂别人的代码
3. 【平时作业、编程小作业】会修改别人的代码，能看懂验证结果
4. 【编程小作业】会修改别人的代码，能手工验证
5. 【编程大作业】会选择、并使用合适的工具进行建模、验证一个模型
6. 【编程大作业】【可选、加分】会改进工具，提升工具的效率
7. 【可选、满分】完全自己编写工具

学习要求

- 平时作业、编程小作业——布置后两周内完成并提交给助教
- 期末大作业：自选题 (二选一)
  - 选择一种形式化语言（不限于本课程所教的语言），设计并证明（或验证）一个较大的模型 (自己命题)，并附上文档
  - 【加分】自选一篇论文（如 CCF A 或 B 类论文），完成大致方案的复现，并附上文档
- 评分规则——按大家的完成水平排序（非标准分）