

木兰编程语言重现项目

三年小结

吴烜

为何重现木兰编程语言？

木兰是套壳 Python 吗？

从“换皮”到“套壳”

- “皮”和“壳”的差别
- “壳”是什么？没意义么？

三年前中科院《关于“木兰”语言问题的调查与处理意见》提及 Python 段落

(四) μ Lang以 Python 为基础，在语法上做了一些 C++风格的修改。刘雷团队实现了编译器前端，将 μ Lang的 LR 文法翻译成 Python 的 AST，并在 AST 上进行了对象命名标准化和 lambda 表达式扩展。语言后端和运行环境直接使用 Python 系统。向媒体宣称“木兰”是“完全自主设计、开发和实现的编程语言”属虚假陈述。

编程语言只能“那样”做吗？

- 不可有依赖？
- 没有技术突破就无商业价值？
- 完美才能用？
- 分析器怎么不是手写的？

层层枷锁.jpg

绝知此事要躬行

- 2020年春节前，初见木兰语言，昙花一现：
<https://zhuanlan.zhihu.com/p/265091649>
- 风波结束，疫情开始，木兰重现项目启动

当前复现状态，可持续性

- 项目目标与复现方法
- 代码构成和行数
- 跟随 Python 语法树升级

目标与方法

- 何谓重现：尽量将 `ulang-0.2.2.exe` 的功能复现而不作修改，包括 `bug` 在内
 - 一些差异后述
- 方法：重头实现一遍——从零开始逐步添加各子功能，包括各条语法规则。编写相应测试用例，确保原始可执行文件也全部通过。

代码构成和行数

	逆向	重现项目	重现项目 测试之外 代码
Python	3315	3986	3217
木兰	69	3315	553
Markdown	56	1566	1552

跟随 Python 语法树升级

- Python 3.7 在 2023 年中停止维护
- 曾尝试升级到 3.8: <https://zhuanlan.zhihu.com/p/391093875>