???? AI Coding

??

1. 人类需要掌握高级的架构/设计，底层的工作已经被编译器和AI替代了
2. 目前AI还只能接受明确的任务，如果有循环依赖，嵌套的问题，就是很理想了
3. 如果需要一个比较巧妙的、高度定制的、高性能的实现或者设计，AI还不能很好的实现
4. 如果需要一个基本功扎实，执行力高，速度快，那么AI将是完美的给你打下手的助理

???????Agent????????????

1. 编程具有容错性，不像自动驾驶等，要求准确度很高
2. 比自然语言更严格的结构化表达，使得**LLM**更擅长于编程
3. 软件工程领域在过去数十年间积累了极其完善的数字化工具链，编译器，CICD，等等
4. 协作模式非常合理，依赖于软件工业的发展，编程协作非常成熟，让AI编程能快速嵌入到工作中，并不断正向迭代
5. 大量的低维度训练数据，归整的数据集
6. 标准的输入输出，可以构造出足够复杂，但是又容易理解的，低维度的归整的输入
7. 编程的方法学比较简单
    1. bug 的寻找，测试编写，都有固定的开发流程
    2. 所有的问题都在代码里面，相关的代码都能放到上下文里面，代码的知识都存在模型权重里面

??

1. 当用户习惯于直接接受AI 生成的代码而不加审查时，代码库的复杂性并没有消失，只是被隐藏了。
2. 高层级的软件架构，大型软件的架构和高层级的抽象，是LLM难以克服的障碍
    1. 大型软件的高层级设计思想和架构往往非常的抽象，当前的LLM还难以轻松理解
    2. 当前LLM只能处理上下文以内的信息，上下文不足以包括复杂软件的所有信息
    3. Agent

        还不能通过像人类一样，通过分层、总结、隔离的方式来理解整个工程，虽然人类也不能同时关注太多事物

Revision #8
Created 2025-12-27 08:40:47 UTC by Colin
Updated 2026-04-29 07:34:00 UTC by Colin