

（深圳）

实验设计报告

开课学期： 大三上

课程名称： 计算机体系结构

实验名称：Pin入门及指令依赖距离分析

实验性质： 课内实验

实验时间： 2021.11.15地点： T2507

学生班级： 1901105

学生学号： 190110509

学生姓名： 王铭

评阅教师：

报告成绩：

实验与创新实践教育中心印制

2018年12月

1.借助Pin的API文档，阅读ManualExamples目录下inscount0、inscout1和inscout2工具的源码，分析插桩工具的代码框架和执行过程

答：代码框架：由插桩代码在源代码中插入分析代码，从而在程序执行时，运行分析代码，从而完成分析的过程。

inscount0在main函数中调用了API函数INS\_AddInstrumentFunction，由插桩代码Instruction在每条代码都插入分析代码docount，从而实现插桩，然后执行。所以，inscount0是指令级插桩。

inscount1和inscount2的执行过程类似，只不过是轨迹级插桩，插桩代码为Trace，分析代码为docount，其中Trace会决定并不是在每一条指令前都插入分析代码，而是在一个基本块前插入分析代码，从而完成轨迹级插桩。

2.a. 可以发现，指令依赖距离分布图存在很长的尾巴。试分析尾巴可能是由哪些寄存器引起的？为什么？

答：“尾巴”即指令依赖距离较长的寄存器，有可能是段寄存器和保存返回地址偏移量的寄存器。

b. 设有2个不同架构的处理器平台，现分别于其上运行由相同版本编译器所编译的测试程序，并分别在这两个平台上使用insDependDist工具对该测试程序进行插桩分析，得到如图6-1所示的指令依赖距离分布图。请问架构A、B之中谁具有更多的寄存器？为什么？

答：B有更多的寄存器，因为A的指令依赖距离中为1的比例要大于B，B平均的指令依赖距离更长，说明B有更多的寄存器。

c. 现有基于相同ISA（Instruction Set Architecture）设计的架构A和架构B。若架构A采用停顿法解决流水线数据冲突，架构B则采用数据转发法，当二者执行相同的测试程序时，它们的指令依赖距离分布图是否相同？为什么？

答：不同，停顿并不会改变指令依赖距离，而采用数据转发，会将ALU计算的中间结果传递给下一条指令作为输入，从而降低指令依赖距离。