



实验2：线程与同步

李辉

hui@xmu.edu.cn



实验2

➤ 实验目的

- 实现Nachos的同步机制：锁和条件变量，并利用这些同步机制实现几个基础工具类

➤ 实验内容

- 实现锁机制和条件变量（60%）
- 实现一个线程安全的表结构（20%）
- 实现一个大小受限的缓冲区（20%）



建议步骤1

- 以 “sleep+中断禁止与启用” 实现锁
 - 首先，思考锁 **Lock** 的类定义里应该包含哪些成员变量（可参考已在synch.cc提供的**Semaphore**的类定义）
 - 实现简单的 **Lock::isHeldByCurrentThread** 函数（提示：在**Lock**的定义中添加适当的成员变量，则函数体内一个语句足矣）
 - 思考上锁的步骤并实现 **Lock::Acquire** 函数（某些步骤可参考**Semaphore::P**函数的实现）



建议步骤1

- 以 “sleep+中断禁止与启用” 实现锁
 - 思考开锁的步骤并实现 `Lock::Release` 函数
 - 测试该锁是否正确实现



建议步骤2

- 以“sleep+中断禁止与启用”实现条件变量并用实验一实现的双向链表测试其是否正确实现
- 以“信号量”实现锁和条件变量并用实验一实现的双向链表测试其是否正确实现
- 选用一种同步机制实现一个线程安全的表结构，并设计程序测试验证
- 选用一种同步机制实现一个大小受限的缓冲区，并设计程序测试验证



注意事项

- 可以阅读 `nachos-3.4/code/synchlist.cc` 和 `nachos-3.4/code/synchlist.h` 以加深对锁以及条件变量如何使用的理解



注意事项

- 关于条件变量的实现，要求采用**Mesa**语义，而非**Hoare**语义
 - 两者什么区别？[参考资料](#)
 - 简单举例：假如有两个线程**T1**和**T2**，**T1**因为资源不满足进入了等待队列，这时**T2**释放了**T1**所需的资源，这时会发生什么呢？**T1**应该被立即唤醒，还是等**T2**结束呢？这种情景下的不同操作区分了两种语义



注意事项

- 条件变量一定要和某个特定的锁变量配合使用，任何进程在调用条件变量的相关方法（**Wait**, **Signal** 和 **Broadcast**）前应先持有锁，所以程序在实现这些方法时必须先用 **isHeldByCurrentThread** 作出判断
- 更多注意事项详见课程网站中 **Nachos** 实验二任务书的相关说明



本次实验提交的特别之处

- 锁机制和条件变量需用“Sleep”和“信号量”两种方式实现
 - 当用Sleep实现，为了提交，这部分实验完成后，将synch.cc和synch.h分别另存为synch-sleep.cc和 synch-sleep.h
 - 当用信号量实现，为了提交，这部分实验完成后，将synch.cc和synch.h分别另存为synch-sem.cc和synch-sem.h
- 实验第2，3部分所需的头文件 BoundedBuffer.h 和 Table.h 在课程网站的lab2-headers.tar.gz文件内



实验2：提交文件

Makefile.common

main.cc

threadtest.cc

dllist.h

dllist.cc

dllist-driver.cc

synch-sleep.h

synch-sleep.cc

synch-sem.h

synch-sem.cc

Table.h

Table.cc

BoundedBuffer.h

BoundedBuffer.cc

nachos02.doc // 实验报告



实验2：提交截止时间

- 实验报告及文件提交截止时间分别为2021年5月13号中午12点
 - 实验报告和代码以组（每组1-4人）为单位打包发给课代表，再由课代表统一发给老师。
 - 是否及时提交以截止时间前课代表是否收到为准。如在截止时间前未提交，则该部分实验报告没有分数。
 - 实验报告（word或pdf）应包含以下内容：小组成员姓名学号、实验报告分工、实验设计和关键代码实现、实验结果讨论、遇到的问题、收获和总结。



实验2

- 实验二工作量比实验一大，希望大家尽早开始。
- 实验检查：5月13日（实验二）课上对每位同学单独进行，老师和助教会就实验报告和代码进行提问，测试你的理解程度。
 - 评分标准：completeness, correctness, programming style, and thoroughness of testing



实验2一些可能遇到的问题总结

- “我把Lab 2提供的头文件加入nachos，测试了下，就报错了”
 - 中文任务书有提到Table.h的构造函数声明有问题，检查了吗？
- “我在用Semaphore实现的时候。。。关中断。。。 ”
 - 用Semaphore实现锁机制和条件变量时，不应该涉及到关中断操作！请仔细阅读英文文档3.2.1节。
- “怎么进行关中断？”
 - 请阅读synch.cc中Semaphore的P操作的实现里的注释



一些问题总结

- “Condition中的wait和signal要实现什么？”
 - 请阅读synch.h中Condition类上部的详细类注释
- “为什么要关中断？”
 - 请阅读synch.cc中Semaphore的P操作函数上的函数注释，理解atomically的含义，同时反思下理论课部分是不是没有好好听课



一些问题总结

- “为了确保安全，我在XXX和XXX处也加上关中断，测试样例也显示实现是正确的”
 - 需要思考为什么要在某处关中断，它在这里是必须的吗？回避无用的关中断。阅读英文文档3.2.1节：However, you may lose points for holding interrupts disabled when it is not necessary to do so. Disabling interrupts is a blunt instrument and should be avoided unless necessary.



一些问题总结（沟通）

➤ 一些不可能得到你想要的回复的问题

– “对于函数XXX，我是先做A，再做B？还是先做B，再做A？”

- ▣ 很多函数只有几步，如果直接告诉你怎么做，就失去了做实验的意义了。这种问题得到的回复一定是“可以思考下为什么XXX”，或是“两种都试一下，然后在测试代码那，看看哪个会出错，再思考下为什么”。
- ▣ 对于这种情况卡住了，那什么样的问题能得到很好的回答：“我试了XXX，但是遇到XXX问题，根据我的理解XXX，能否给我解释下为什么”。实验任务是用来帮助你理解理论课的内容的，希望能看到有进行思考，否则在实验检查的时候你可能回答不上来。



一些问题总结（沟通）

- 一些不可能得到你想要的回复的问题
 - “我们完成了任务2，但是实验结果XXX那不对，能否帮我们调试下”（将所有代码打包发来）。
 - ▣ 如果老师帮你们直接调试完了，那是谁做完了这个实验？
 - ▣ 比较好的提问方法：“实验结果XXX那不对，我们觉得可能是这里有问题（截图这部分代码片段），但是按我们的理解XXX，这是为什么？”



总结

- “实验二工作量比实验一大，希望大家尽早开始”
 - 不是代码量大，而是思考量大，通过任务2可以帮助大家加深对理论课知识的理解
- 一定要阅读代码里的注释！
- 避免无效沟通
- 多和同组同学讨论，确认下任务书和英文文档中有没说明了，如果没有再询问老师



谢谢
