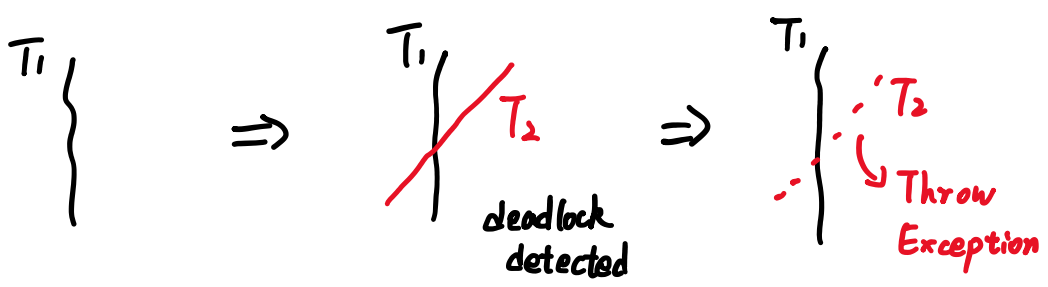
Lab5调试若干Tips

2019年6月12日 星期三　　12:40

关于死锁的行为

当两个或多个不同 transaction 发生死锁时，其中的一个或多个的执行会抛出 TransactionAbortException. 测试程序会 catch 并重启/放弃这个 transaction. 在调用 transactionComplete（commit=False）后，应仿佛什么都未发生过。



关于 BTreeTest 测试

测试程序会创建上千个线程，每个线程会做 插入/删除等操作。每个线程对应一个事务，且结束过的事务可能重用编号. 每个线程会 catch 判断是否成功，并更新结果；最后会 search 确认事务的实现是否正确.

几个可以自测的方式

1. 把测试用的线程用 synchronized 锁在 insertedTuple 上，使事务之间不互相交错，可验证 B 树实现是否正确
2. 可把所有读锁暂时改为读写锁（exclusive），因为占用树根结点，会避免死锁出现，可验证读写锁实现是否正确
3. 可将死锁信息输出，确认死锁是否按预期出现.（比如全部只读操作期间不应出现死锁）

多线程 B 树报错怎么办（找不到键等）

很可能是由于 ① commit 过的事务未正常提交
② abort 的事务未正常撤回

（特别注意，如果使用 set/getBeforeImage 来做恢复，记得 flush 的时候更新）

Synchronized 怎么用？什么时候用？

参照官方文档，可以特别关注下 guarded lock 章节

一般在访问非线程安全的成员变量以及 可能多线访问的文件读写时需要加 Synchronized

我怎么知道我的程序还在跑，还是死锁了？

~~多等会儿~~ 使用 Intelij 的同学可以通过测试运行期间左下角的相机图标查看 dump（各线程是 RUNABLE/BLOCKED/WAITING），如果没有线程 RUNABLE 就是锁死了...