Exercise #4A-1 (Individual)
CS361
Concurrent Programming

Your name: Dennis George

(8 minutes)

1. The course so far has discussed several issues that can affect program correctness caused by "surprises" that occur when multithreaded programs are in operation. Give these issues names, explain their causes, and explain what the course has suggested as ways of working through them. A phrase for each issue, cause, or remedies is all that is necessary for this exercise. We get at least five or six phrases to give an extensive treatment of this, but you may have more.

Interleaving - This can happen when multiple threads are running the same instructions. There are many different possible states that a parallel program can get in to. One solution is to use locks to prevent multiple threads from entering a critical section of code.

Race condition - A race condition is when multiple threads access a shared variable. The outcome of your program can be completely unexpected if two separate threads are executing instructions simultaneously. Some checks in your program may not have the values you would expect. In order to prevent race conditions, you may have to consider the design of your program. Race conditions are hard to test for

Caching coherency - When multiple threads are sharing access to a variable, the CPU caches could fall out of sync with writes and reads. In order to solve this, you can use Java's 'volatile' variable attribute, can guarantee the visibility of variables across threads.

Java optimizing compilers - Optimizing compilers feel free to change the order of code in the procedure when there is no dependency between the variables in a method. The optimizer doesn't necessarily consider the effects of instruction reordering on other threads' variables. This is something that has to be solved by design.

2. Fill out a table that compares how these Boolean operators work (3 points)

A	B	A ∧B	A ∨ B	A ⇒ B
True	True	True	True	True
True	False	False	True	False
False	True	False	True	True
False	False	False	False	True

Of course the point of doing this is not just to fill out the table. We want you to provide more mental organization about the properties of these operations.

Fill the blanks:

A ⇒ B is true except when A is True and B is False.

A ∧ B is false except when both A and B are True.

A ∨ B is True except when both A and B are False.

Scoring for this exercise: +1 point for a reasonable attempt at answering all the questions there is available time for.