CS455 - Lab 04

Thilina Buddhika

February 23, 2015
Course logistics
Quiz 4 review
Term paper deliverable 01
ASG 02 - FAQ
Course Logistics

1. Term Paper deliverable 01 is due this Friday by 5:00 p.m.
2. HW1-PC grades will be released during this week
3. Check and verify the grades on RamCT
4. Report about group members who are not cooperating
1. A program that has concurrency bugs may continue to function correctly if the rate of invocations and the number of threads are below a certain threshold. [True/False]

2. A thread-safe class does not require any additional synchronization or coordination by the calling class. [True/False]

3. Storing state variables of a particular class in public fields allows other classes within that program to reason about thread-safety. [True/False]
4. The `wait()`/`notify()` mechanism in Java has an inherent race condition that cannot be solved without deep integration with the JVM. [True/False]

5. The key to thread-safe programming is not so much what the object does, but rather how it will be accessed. [True/False]

6. If two operations executed by threads are atomic with respect to each other, each of these operations should map to a single underlying machine level instruction that is executed atomically. [True/False]
7. Stateless objects are always thread-safe. [True/False]
8. Consider a variable count of type long. If the mutation operation on this variable is the increment operator (++) there is no need to synchronize accesses to the mutation operation. [True/False]
9. We only need to synchronize accesses to write operations on a variable. The read operations need not be synchronized. [True/False]
Which of the following constructs does not address the memory visibility issue in concurrent programming?

a) Use of the volatile keyword
b) Use of the synchronized keyword
c) Using the AtomicInteger and AtomicLong variable types
d) Using the final variable
Significant progress on:

- Introduction [500 words]
  - the importance
  - where is it used, use cases, etc.

- Problem characterization [500 words]
  - what makes it harder to solve the problem?

- Trade-off space for solutions in this area [400 words]
  - Each (general) solution to the problem has their pros and cons.
Significant progress on:

- Dominant approaches to the solution [1000 words]
  - Detailed analysis of each solution
  - Do not defend the authors
  - Provide citations.
Thread Pool Manager

- Launch a preconfigured number of worker threads during the initialization
- Maintains two queues.
  - Tasks
  - Available Worker Queues
- Runs on a separate thread, continuously assigns tasks to an available Worker thread.
- Implement wait-notify.
  - Wait until a task/worker is available
  - Notify once a task/worker is available.
Abstract representation of a job carried out by a worker thread
- Crawling a web page/ task hand-off
- May produce many child tasks.
A Runnable object
Wait till a task is assigned by the Thread Pool Manager.
Executes a task
Returns the resulting tasks back to the Thread Pool Manager.
Returns itself back to TPM as an available worker.
**ASG 02 - FAQ**

- Implement a filtering and pre-processing scheme for URLs
  - Remove links for extensions .pdf, .doc, .ppt, etc.
  - Process relative URLs.
- **Sleep 1s after crawling a page.**
  - Do not overload the servers.
- **ASG 02 writeup is updated.** Fixed an inconsistency in the sample configuration file.
Take an iterative approach to develop.
- Break the entire assignment into smaller functionalities.
- Understand the dependencies.
- Implement a functionality and test it before proceed to the next functionality.

Please follow the package, class naming conventions and the packaging instructions.
- Many of you did a good job in ASG 01.
Wrap Up

Questions ?