Agenda

- Course Logistics
- Quiz 3 review
- ASG 02
- **ASG-01 Programming component** is due Wednesday, 18th Feb. by 5:00 p.m.

- **ASG-01 Written component** is due Friday, 20th Feb. by 5:00 p.m.

- **Term Paper - Deliverable 01** is due Friday, 27th Feb. by 5:00 p.m.
  - Around 2000 words.
  - Significant progress on Introduction, problem characterization, trade-off space for solutions in the particular area, dominant approaches to the problem.
  - A bibliography of at least 10 references.
  - Any problems that you are having with your non-performing teammate should be reported at this time.
1. Consider a class A that has two synchronized methods s1() and s2(); this class also has two unsynchronized methods u1() and u2(). Class A was used to create two object instances, a1 and a2, in a particular process P. Within the process P, there are N threads that are represented as T1, T2, . . . , TN. Please indicate if the following 5 statements are True/False. Please circle your choice.

a) Threads T1, T2, . . . , TN can all be active in instance a1 at the same time and have different program counters. [True/False]

b) Threads T1 and T2 can be active inside method a1.s1() at the same time. [True/False]

c) Thread T1 can be active in a1.s1() and Thread T2 can be active in a1.s2() at the same time. [True/False]

d) Thread T1 can be active in a1.u1() and Thread T2 can be active in a1.u2() at the same time. [True/False]

e) Thread T1 can be active in a1.s1() and Thread T2 can be active in a2.s2() at the same time. [True/False]
2. The scope of a lock impacts the degree of concurrency for threads within a process. [True/False]

3. Java uses the same mutex lock to ensure thread safety for both static synchronized methods and non-static synchronized methods. [True/False]

4. The `volatile` keyword for a variable is used for performance reasons because it allows Threads to cache that variable in a register. [True/False]

5. Using the `volatile` keyword ensures that increment (++) and decrement (--) operations on an integer variable are atomic and thread safe. [True/False]
Which of the following statement is incorrect about wait-and-notify in Java?

a) It has an inherent race condition that is solved by the JVM.
b) It can be used as a communication mechanism between threads.
c) wait() and notify() methods are inherited from the Object class.
d) It can be used instead of synchronization to achieve thread safety.
ASG 02 Discussion

- Implementing the thread pool
- Tasks
- Worker Threads
- Using the external HTML parser
- Task hand-off
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.
ASG 02 - Task

- 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.
// disable verbose log statements
Config.LoggerProvider = LoggerProvider.DISABLED;

try {
    // web page that needs to be parsed
    final String pageUrl = "http://www.cs.colostate.edu/~cs455";
    Source source = new Source(new URL(pageUrl));
    // get all 'a' tags
    List<Element> aTags = source.getAllElements(HTMLElementName.A);
    // get the URL ('href' attribute) in each 'a' tag
    for (Element aTag : aTags) {
        // print the url
        System.out.println(aTag.getAttributeValue("href"));
    }
} catch (IOException e) { // in case of malformed url
    System.err.println(e.getMessage());
}
Each domain/root url should be handled by a separate crawler.
Crawlers communicate with each other.
A crawler should parse only the pages from its domain.
Every crawler should run on the same port on different hosts.
Resuse the transport implementation from assignment 1.
Tasks handed off by another crawler are added to the task queue and the recursion level should be set to 0.
Meta-data about crawlers are provided through a configuration file.

denver:18080,http://www.bmb.colostate.edu/
austin:18080,http://www.cs.colostate.edu/
atlanta:18080,http://www.chm.colostate.edu/

Port number should be determined based on your CSU ID to reduce the chances of possible port conflicts.
Questions ?