

In-Class Exercise: Java Threads

Instructors: Mackale Joyner

Course Wiki: <http://comp322.rice.edu>

Staff Email: comp322-staff@mailman.rice.edu

Goals for today's exercise

- Experimentation with Java threads

This in-class exercise can be downloaded from the following svn repository:

- https://svn.rice.edu/r/comp322/turnin/S20/NETID/inclass_ex

Use the subversion command-line client or IntelliJ to checkout the project into appropriate directories locally.

In today's exercise, you need to use NOTS to run performance tests.

1 Conversion to Java threads: Spanning Tree

1. The `SpanningTreeSeq.java` program is an example sequential solution to the spanning tree problem. The `SpanningTreeAtomicHjLib.java` program is a provided parallel solution to the minimum spanning tree problem. This version uses `finish` and `async` constructs along with an `AtomicReference`.
2. Your task is to convert `SpanningTreeAtomicHjLib.java` to a Java program that uses threads instead of `HJlib` tasks. You should modify the provided `SpanningTreeAtomicThreads.java` file, and use Java thread methods instead of `finish/async`. (The `AtomicReference` calls can stay unchanged.) There are `TODOs` in the file to guide you.
3. You have been provided with tests for your parallel spanning tree implementation in `SpanningTreePerformanceTest`. To complete this portion of the lab, you should submit these performance tests to NOTS by modifying the provided `myjob.slurm` template and submitting manually.

2 Programming Tips and Pitfalls for Java Threads

- Remember to call the `start()` method on any thread that you create. Otherwise, the thread's computation does not get executed.
- Since the `join()` method may potentially throw an `InterruptedException`, you will either need to enclose each call to `join()` within a *try-catch block*, or add a *throws `InterruptedException`* clause to the definition of the method that includes the call to `join()`.

3 Testing on NOTS

Now that we an implementation of a spanning tree using threads, we'll measure the actual performance of the thread implementation without interference on your laptop.

To do so, you should use the provided `myjob.slurm` file. As usual, when using the `myjob.slurm` file please open it to fix any `TODO` items.