CSC 563 Fall 2013

Homework 3
Due: Thursday 10.10.13

Exercise 1 – thread coordination

Write a multithreaded program that models the following behavior. 10 threads write random integers in the range 1—20 to an array of 10 integers. A thread is may only write to the array location corresponding to its ID (thread 0 writes to array index 0, etc.). Each thread is assumed to only be able to see neighboring array values with the exception of threads 0 and 9 which can only see one neighboring value.

After each thread writes a random value to the array, it updates a global count of writes and checks neighboring locations to determine if the values form a sequence of 2 or 3 consecutive integers (e.g. thread 5 generates 12 and sees 11 at location 4 and 13 at location 6). If such an arrangement is found, the thread sets a flag and signals a global thread. If not, it generates a new random value and checks neighboring values again. The global thread determines if the array values form a sequence of consecutive numbers, in which case it prints out the array contents and the total number of writes before the program terminates. If that's not the case, the thread clears the flag and waits for a thread to signal it.

Exercise 2 – a simple search engine

Write a multithreaded search engine that processes the contents of all top-level URLs referenced by a given URL (limited to http only). The program takes for input a URL and a search string as command-line arguments and outputs all occurrences of the search string in the top-level html files referenced in the URL. Each of the html files should be retrieved by a thread managed by an Executor object with a fixed thread pool.

Submission

Deposit your source code in the Google folder that you previously shared by midnight of the due date.