CSCI 415: Networking and Parallel Computation

Assignment #2: Multi-threading

Deadline: Tuesday, October 3^{rd} at 11:59PM

The objectives of this assignment are:

1. Understanding the steps involved in designing a parallel program using Multithreading.
2. Understanding using locking mechanism (e.g., mutux) for accessing critical sections.
3. Understanding using barrier synchronization.

1 Description

In this assignment you will write a multi-threading program (in C++) that computes the clustering coefficient of a graph. Multiple graphs are attached to the assignment (toyGraph.txt, toyGraph1.txt, and HcNetwork.txt).
Clustering Coefficient: The clustering coefficient of a node v_i is a measure of the density of edges in the neighborhood of the node v_i. The neighborhood of a node v_i is denoted as G_i which is the graph induced by v_i and its neighbors. The clustering coefficient of a node v_i is defined as

$$C(v_i) = \frac{\text{\# of edges in } G_i}{\text{maximum number of edges in } G_i} = \frac{2 \times m_i}{n_i \times (n_i - 1)}$$

where m_i is the number of edges in G_i and n_i is the degree of node v_i. m_i is simply the number of edges that exist between the neighbors of v_i. If a node has a degree less than 2, assume that its local clustering coefficient is 0. The clustering coefficient of a graph G is the average clustering coefficient over all the nodes of the graph.

$$C(G) = \frac{1}{n} \sum_{i=1}^{n} C(v_i)$$

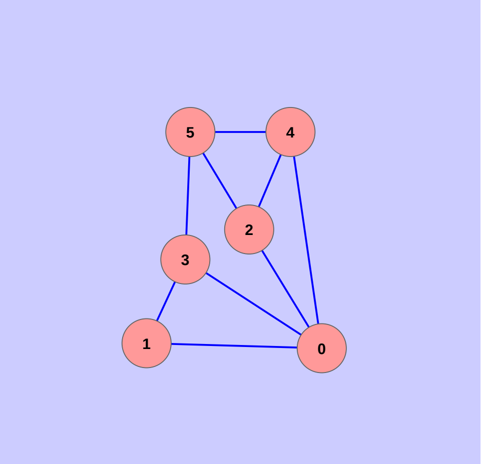

Figure 1: Example Graph

Example 1: In Figure 1, the clustering coefficient of node 0 is $\frac{2}{6}$ since there are only 2 edges among the neighbors of node 0. Node 0 has 4 neighbors and thus there could potentially be 6 edges $((\binom{4}{2} = (4 * (4-1))/2))$ among these neighbors.

Example 2: the clustering coefficient of node 5 is $\frac{1}{3}$ since there is only an edge among the neighbors of node 0 (nodes 2, 3, and 4). Node 5 has 3 neighbors and thus there could potentially be 3 edges $(\binom{3}{2} = (3*(3-1))/2)$ among these neighbors.

For the graph shown in Figure 1, the clustering coefficient of the graph is computed as follows:

$$C(G) = \frac{1}{n} \times (C(0) + C(1) + C(2) + C(3) + C(4) + C(5))$$

$$C(G) = \frac{1}{6} \times (\frac{2}{6} + \frac{1}{1} + \frac{2}{3} + \frac{1}{3} + \frac{2}{3} + \frac{1}{3})$$

$$C(G) = \frac{1}{6} \times \frac{20}{6} = 0.555$$

While implementing the algorithm, each thread shall acquire a lock before updating the global sum variable. You have to make sure that all the threads are done working before finding the clustering coefficient of the graph. This can be achieved using barrier synchronization.

2 What to turn in:

Submission: You should submit your code along with a readme file (.txt,.docx, or .pdf) explaining how to run the program. In your readme file, report the running times for various number of threads: 1, 2, 4, 8, and 16 on the graph in 'HcNetwork.txt'. The file should be named in the following format, useridAssig2.zip, .tar, or .tgz. Late submission will get a 10% penalty for every late day.