Final Project
CSC 311: System Software
Fall 2019 (Block 4)

Due Date: Wednesday, December 18, 8:55 AM

Points: 100

Submission: A compressed folder containing your source code, readme file, and a pdf file containing your project report.

Description:
For this group project, you have work in a group to design and implement the solution to the following problem.

Problem Statement:
Design and implement a distributed (centralized or decentralized) load balancing system for homogeneous cluster.

Explanation:
You have to design and implement a system which performs following tasks:

1. **Workload Monitoring System:**
   This system is responsible for monitor the CPU utilization of each node (computer) in the cluster. If you are designing centralized system, then each node should send its utilization information to the central server (master). If you are designing decentralized system, then each node should send its utilization information to every other server. The aspects you need to focus on are:
   a. How to get CPU utilization of a node.
   b. How can a node send its utilization information to another node?
   c. Once node A received an information from node B, how does node A identify that the information is sent by node b and not by node c?
   d. How does a master (in case of centralized system) or a node (in case of decentralized system) stores the utilization of all the nodes in the system?
   e. How often this information transfer should take place?
   f. What is impact of the choices you have made for the above-mentioned questions on the resource utilization of the entire system?

2. **Job Submission System:**
   This system should provide a front end (command line) for the user to submit a job in the form of an executable file and at max one input file.

3. **Load Balancer System:**
   Upon job submission, this system should select a node in the cluster to execute the job based on the utilization information of all the nodes in the system and the load balancing strategy designed by you.

4. **Dispatcher System:**
   One the load balancer selects the node to execute the job, dispatcher should transfer the executable file as well as input file (if provided) to the selected node. Execute the job on that node, collect the output in a file, and returns the files to the master node.

Submission:

1. **Source Code:**
   This part should contain one or more files with .c or .h extensions.

2. **Project Report:**
   This should not be longer than 4 pages. It should focus solely on the design of your system and the performance implications of your design choices. You should use ACM conference template to create this project report.

3. **ReadMe file:**
   This file should contain short and precise description of major source files in your source code. It should also specify how to compile your source code and how to run your system.

Resources:

Please refer to the Moodle assignment created for this project for the resources.