

Assignment 2 (10%)

Date Given: Feb 15, 2020

Submission Due: Feb 28, 2019 at 11:59 pm (midnight)

**** Late submissions are not accepted and will result in a 0 on the assignment**

Objective:

This assignment covers cloud computing tools, and frameworks that can be used for performing parallel operations.

Grading Scheme:

- Section A: 10%
- Section B: 30%
- Section C: 15%
- Section D: 20%
- Section E: 20%
- Adding citation in IEEE/ACM Format only. Use reliable information source: 5%

Academic Integrity:

- This assignment does not require group work. Therefore, each student is expected to complete their work by themselves. Collaboration of any type amounts to a violation of the academic integrity policy and will be reported to the AIO.
- Do not copy texts verbatim from online or printed materials
- Do not copy texts from other's work
- Do not submit other's work
- If you obtain help from Tutor(s), please acknowledge
- Provide citation for texts, images, tables, data etc.
- The Dalhousie Academic Integrity policy applies to all material submitted as part of this course. Please understand the policy, which is available at: https://www.dal.ca/dept/university_secretariat/academic-integrity.html

Hypothetical Scenario:

Consider "Cloud5409" is a cloud service provider company that provides a scalable cloud infrastructure, and scalable service(S). The company is developing an application, which performs two tasks using two services: S_1 , S_2 . Presently, the company focuses on scalable architecture. Once it is developed "OpenStack" will be used to design the private cloud. [Note: Assignment 2 does not require you to use OpenStack, it is written just to present the scenario]

S_1 performs the following using Framework₁:

- Factorial calculation of given number N, (e.g. $N=5$, result= 120)
- Stores the request ID, time, N, result in a log file, which serve as a state database

S_2 performs the following using Framework₂:

- Fibonacci calculation of given number N (e.g. $N=5$, result=0,1,1,2,3)
 - Stores the request ID, time, N, result in a log file, which serve as a state database
- ✓ The value of "N" must be fetched from a text file (input.txt). For the value of N, you need to store 100 random integer numbers. Each line in the input.txt will represent one N value.
- ✓ Framework₁, Framework₂ runs in parallel.

Follow the steps to perform the above-mentioned operations:

Section A: To setup the server, you need to download and install Apache Mesos cluster in your system. The documentation is available in <http://mesos.apache.org/documentation/latest/>

Section B: Perform background Research on Mesos, and for that read the following conference paper published in IEEE. You need to write a 1-page summary on the paper (Font Type=Times New Roman, Size=11). [Do not copy images, or text verbatim]. **In addition, please include insightful comments based on your understanding.**

P. Saha, A. Beltre and M. Govindaraju, "Exploring the Fairness and Resource Distribution in an Apache Mesos Environment," *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, San Francisco, CA, 2018, pp. 434-441.

Section C: Explore and install Marathon framework on Mesos cluster. Follow the documentation available in <https://mesosphere.github.io/marathon/>

Section D: Create a GitHub repository for your current project, e.g. "newProj5409".

- Simulate a behavior of 2 developers. Install **git** tool in your system.
- Clone the project to two different folders in the system
E.g. /Documents/ user1/newProj5409
E.g. /Documents/ user2/newProj5409
- Create one program file for factorial, and store it to a local git folder.
E.g. /Documents/ user1/newProj5409
- Add the factorial program file to the repository
- Get the file from repository to another user's folder E.g. /Documents/ user2/newProj5409
- Create another program file Fibonacci series program, and store it in another local git folder
E.g. /Documents/ user2/newProj5409
- Add the files to the repository

Section E: Using Marathon launch two tasks in two frameworks. One task should perform Factorial operation, and another task should perform Fibonacci series computation. Measure the time taken by each task, and generate a graph. (X axis = N value, Y axis = Time)

Submission Instruction:

- Create a Folder with your name and B00 number, and store all your files –
 - Program Source code with readme and/or libraries
 - PDF file containing all answers
 - Screenshots and image files (if any)
- Compress the folder and create a .ZIP file (do not use other compression formats)
- Upload the .ZIP file on Brightspace.
- Submission Due: **Feb 28, 2020 at 11:59 pm (midnight)**