**Developing Soft and Parallel Programming Skills Using Project-Based Learning**

Fall 2018, Group 5

Austin Nocero, Rickey Clark, Paige Park, Nicholas Economou, Zhiyi Dong

***Catalog----***

***Planning And Scheduling***

***Parallel Programming Skills***

***Appendix: Content Links***

**Planning and Scheduling:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Assignee Name** | **Email** | **Task** | **Duration (Hours)** | **Dependency** | **Due Date** | **Note** |
| Nicholas Economou | neconomou1@student.gsu.edu |  | 2 hours |  | 11/30/18 | Must be ready 24 hours before the due date |
| Paige Park | ppark11@student.gsu.edu | GitHub and Task 3 | 3 hours | Lab reports | 11/30/18 | Must be ready on the first day of group work |
| Zhiyi Dong | zdong4@student.gsu.edu |  | 3 hours |  | 11/30/18 | Must be ready 5 hours before deadline |
| Austin Nocero  (Coordinator) | anocero1@student.gsu.edu | Task 3 and Written Report | 3 hours |  | 11/30/18 | Must be ready 5 hours before deadline |
| Rickey Clark | rclark39@student.gsu.edu |  | 3 hours |  | 11/30/18 | Must be ready 5 hours before deadline |

**Parallel Programming Skills:**

***Foundation: Use the reading material to answer the questions (in your own words)***

***Question 1) What are the basic steps(show all steps) in building a parallel program? Show at least one example.***

There are four steps in creating a parallel program: Decomposition, Assignment,

Orchestration, and Mapping.

Decomposition: break up problem into tasks that can be carried out in parallel and

the key aspect of decomposition is identifying dependencies.

Assignment: think of the threads as workers. The goal is to balance workload,

reduce communication costs, and it can be performed statically, or dynamically

during execution.

Orchestration: involves structuring communication, adding synchronization to

preserve dependencies, organizing data structures in memory, and scheduling

tasks. The goal is to reduce costs of communication, preserve locality of data

reference, and reduce overhead.

Mapping: execution units. Parallel application uses the entire machine, so

oversubscribing machine with multiple parallel apps is not common.

Example: PI Calculation

- Divide the loop into equal portions that can be executed by the pool of tasks

- Each task independently performs its work

- A SPMD model is used

- One task acts as the master to collect results and compute the value of PI

***Question 2) What is MapReduce?***

MapReduce is a programming model that provides massive scalability across thousands

of servers in a Hadoop cluster. MapReduce is considered as the heart of Hadoop. The

term MapReduce actually is the merging form of two different words map and reduce.

this means that there are two different and separate task which Hadoop performs.

***Question 3) What is map and what is reduce?***

Map job converts the set of data and converts them into another set of data. where

individual elements are broken down into value or key pairs.

Reduce job is always performed after map job, following the sequence of MapReduce.

The Reduce job takes the output from a map as input and combines those data tuples

(value or key pairs) into smaller set of tuples.

***Question 4) Why MapReduce?***

MapReduce is a programming model using which we can write applications to process

huge amounts of data, in parallel, on large clusters of commodity hardware in a reliable

manner.

***Question 5) Show an example for MapReduce.***

Count of URL Access Frequency: Map job processes web log and outputs <URL,

1>. Reduce job emits the percentage of total access for that URL <URL, total

count>.

***Question 6) Explain in your own words how MapReduce model is executed?***

To begin the data is automatically split into sets of what are called M splits. All of these splits that hold this data then can be processed in parallel. Copies of the program then run with one of them being the master and the rest of them being the workers. Then there are map tasks and reduce tasks which are assigned to the workers by the master. The workers that are given the map task then reads the data off of the splits and gives key pairs which is assigned to memory. These keys are then written on the local disk where they are given to the master which then gives them to the assigned reducers. The reduce workers then read these keys and assign the data correctly so that the corresponding data goes together. The outputs of the reduce workers then goes to a final file. When it is complete the master goes back to the users program and MapReduce ends and the program goes back to the users code.

***Question 7) List and describe three examples that are expressed as MapReduce computations.***

First example is count of URL access frequency. The map function gets the workers to get the number of requests from a web page and then the reduce workers adds them all together so that you can get a total. The second example is term-vector per host. This is where the map function takes a file and gets term vectors, then the reduce workers add the term vectors together, removing infrequent terms, and shows a final pair. Finally, the third example is distributed grep. This is where the workers will emit a line if a pattern that is given is matched, then the reduce workers copies that data and combines it to an output.

***Question 8) When do we use OpenMP, MPI and, MapReduce(Hadoop), and why?***

We use OpenMP, MPI, and MapReduce when we want to add the parallelization feature into your data. OpenMP is good for when you have functions in your code that you want to add parallelization. The function is then split into multiple threads where the data is handled by each. MPI is good to use when you want to use parallel code on more than one machine. MapReduce is very good to use for when you have large amount of data that you want to process. MapReduce does not perform very efficiently when used with smaller amounts of data.

***Question 9) In your own words, explain what a Drug Design and DNA problem is in no more than 150 words.***

The problem with Drug Design and how it relates to DNA is how complex proteins are. In proteins there are very small molecules called ligands that need to be the correct shape and size to fit a particular protein. So, when we are testing a ligand, we have to test multiple orientations to make sure the ligand binds in a way that fits the protein in a useful way. Then we have to identify the particular ligand that scored the highest and matches up with the protein in a way that produces the results that we want. So the problem is finding the best ligand for the situation and how software can develop these different ligands to test.

**Appendix: Contact Links**

Slack:<https://csc3210group5.slack.com/archives/CCLF40G93/p1536007126000200>

YouTube channel:<https://www.youtube.com/channel/UCOP-InfUQx_cPpknqOH4cZA>

Github: https: <https://github.com/Csc3210GroupFive/Group5/projects/5>

