Department of Computer Science and Information Systems
ASHESI UNIVERSITY, BEREKUSO, GHANA

CS434 — Introduction to Parallel and Distributed Computing

CS434: Introduction to Parallel and Distributed Computing

To Be Completed By: 11:59Hrs, February 27, 2020

Outcome

This is a group assignment. We expect to familiarise yourself, learn and understand the use of the following:

i.) The Unix/Linux programming environment. You have a choice of:

- Use the provided Lab resources but boot into Linux.
- Ubuntu 18.04 (installed on your personal laptop by dual booting if your main environment is Windows 10)
- Cygwin within Windows 10, if you do not like dual booting.
- Install and Use the Linux Bash Shell on Windows-10. You must be running the 64-bit Windows-10 and select the Ubuntu distro.

ii.) The use of one or more of the following editors – vi/vim/gvim, gedit, atom or emacs;

iii.) How to run and compile programs with your favourite compilers; a choice of one or more of gcc 6.4.0/7.4., python-3.6.X

iv.) How to use the gcc compiler in a Unix environment to compile and run a sample program successfully that uses the PThread library, i.e. adding ”-lpthread”

v.) How to use the gcc compiler in a Unix environment to compile and run a sample program successfully that uses the OpenMP library, i.e. adding ”-lopenmp -lgomp.” See the sample text below for a ”hello_world.c” code.

```
#include <stdlib.h>
#include <stdio.h>
#include "omp.h"
int main() {
#pragma omp parallel
    {
        printf("hello world \n");
    }
}
/*
    To compile
    % gcc -fopenmp -lgomp hello_world.c

    To run use:
    export OMP_NUM_THREADS=8 # <number of threads to use>
    % ./hello_world
*/
```

vi.) How to write a pseudo-code for relevant modules of your programs. We recommend the use of *Algorithm2e.sty* for your pseudo-code writing [1].

vii.) How to use GIT and setup a group project account on GitHub.

viii.) **Please include a documentation branch for your reports** in GitHub. Your reports are expected to be written in **LaTeX**. You may start simply by checking out this paper [2] Do not play the trick of writing your documents in MSWord and then checking in only the pdf files of the documentation. I'll be *cloning* your entire project directory for the marking.

Please keep in mind that this laboratory exercise and the subsequent ones are intended to prepare you for your projects — mainly writing software/applications that will involve parallel execution, parallel I/O, big data and may be some machine learning. You will be expected to follow-up this exercise here with learning to use other tools, ***only if needed***, like HTML, CSS, JavaScript, JSon, XML, SQL (BerkeleyDB, SQLite, PostgreSQL, MariaDB, etc.), Jupyter Notebook, R, Julia.

Please ensure that you save copies of your laboratory assignment on your **pen-drive** or **flash-drive** as backups **PLEASE CHECK OFTEN THAT YOUR PEN DRIVES ARE VIRUS FREE, and have the most up-to-date backups.**

Work Schedule

Unix Tutorial for Beginners

If you are not already familiar with Unix then consult any Unix tutorial site and drill yourself on how to use this environment. Familiarise yourself with basic commands such as: ”ls, cd, mkdir, pwd, cat, less, etc.”

Problem Description: Indexing Multidimensional Array of Varying Dimensions at Run-tiem

You are required to write 3 procedures in C each of which takes as input a K-dimensional, $1 \leq K \leq 16$, integer array $A[N_0][N_1]...[N_{K-1}]$ and bounds $N_0, N_1, \dots N_{K-1}$, for varying values of K. The first procedure initialises the elements of the array to zeroes, The 2nd procedure sets 10% of the elements of the array uniformly to 1's, and the third, in a uniform random fashion, chooses 5% of the elements of the array and prints the coordinate indices of the elements and the value of the array element at a chosen cell.

You are then to write a main program, that dynamically generates, in turn 4 arrays A[100][100], A[100][100][100], A[50][50][50][50] and A[20][20][20][20][20] and allocates memory when invoked. For each array allocated your main program calls the three procedures to perform the required operations on it. Normally in each of the 3 procedures to be written one would write the code with a nested loop to probe the elements of the array. This will only work if the number of dimensions is static. The key idea then is to come up with a method to step through the elements of a K-dimensional array when the dimensionality or rank of the array varied at run-time.

The Deliverable

- Submit your computer codes, for marking, to your group's initialised repository on GitHub.
- Provide a short description of your solution to this problem including the pseudo-code of your routines.
- Write a short set of instructions on how to access your GitHub repository and submit this to CANVAS. This should be submitted by only one member of your group.

References

[1] LaTeX/Algorithms, “ LaTeX Wikibooks”

[2] LATEX for Beginners