

EECS3221 Assignment A1

You have to work individually. You are not allowed to view or exchange documents with your peers. We treat all sorts of cheating very seriously. Do not copy code from anywhere, even as a "template". No late submission will be accepted. All assignment work must be done on the EECS red server. Your work will be graded based on: i) correctness of programming logic; ii) clarity of your code and your coding style; iii) whether your code follows the specification. It is your responsibility to explain clearly every detail you do in the code with appropriate comments to avoid any possible confusion in marking. This assignment should be implemented using the mechanisms covered up to Chapter 4, inclusive.

You are required to process several data sets and calculate certain values based on their content. Each data sample is a floating point number (use double precision). Each data set consists of many data samples and each data set is stored in a file. The file format is that all data samples in this data set are separated by whitespace in the file. As an example, we have posted three data sets on the course site: `dataset1`, `dataset2`, and `dataset3`, for your reference. However, your program should be able to process any number of data sets with properly formatted data samples.

- Write a C program, called `process.c`, to do the calculation concurrently with multiple processes. The program first reads the file names of all data sets from the command line. For each data set, it creates a child process. Each child process reads the data samples from its file and calculates: i) the sum of the minimum value and the maximum value in the dataset ($sum = min + max$) and ii) the difference between the minimum value and the maximum value in the dataset ($dif = min - max$). Then the child process sends the calculated `sum` and `dif` values to the main process via a Unix pipe. The main process will print to the standard output the file name of the data set followed by the received `sum` and `dif` values and the recalculated `min` and `max` values. The output for one data set occupies one line as shown below. You may structure your code either as i) waiting for all child processes to finish then processing their data in the order the children were created or ii) waiting for any child to finish and processing its data then repeating this until all children are finished (more difficult but may earn you some extra credit). Finally the main program, based on the received values, calculates the `minimum` and the `maximum` (just 2 values) for all samples in all datasets and prints them to the standard output. The output of the main program should be as follows:

```
Filenamei  SUM=sumi DIF=difi MIN=mini MAX=maxi
Filenamej  SUM=sumj DIF=difj MIN=minj MAX=maxj
...
MINIMUM=minimum MAXIMUM=maximum
```

- Write a Pthread C program, called *thread.c*, to do the above calculation concurrently with multiple threads. The program creates a different thread to process each data set. Each thread does the calculations for its data set and passes the results to the main program. The main program waits for the threads to finish (using non-blocking join may earn you some extra credit) and, based on the obtained values, calculates and prints the final results to the standard output in the same format as above.

Your programs should compile as follows:

```
cc -o prs process.c
cc -o thr thread.c -lpthread
```

And the programs should run as follows (all datasets are provided as command line arguments):

```
./prs dataset1 dataset2 dataset3
./thr dataset1 dataset2 dataset3
```

Make sure that you include the following information (please complete) as a comment in the beginning of your C programs:

```
/*
Family Name:
Given Name(s):
Student Number:
EECS Login ID (the one you use to access the red server):
YorkU email address (the one that appears in eClass):
*/
```

Finally prepare your report as a PDF file named `reportA1.pdf`. Your report should be structured as follows:

1. A general outline of your understanding of the assigned work.
2. A clear statement about the assigned work/components you believe you have done/completed successfully.
3. A statement about the work you believe you might have not completed successfully (feel free to comment on related problems, if any).
4. Anything else related to your work that you might wish to comment upon.

Please copy the above entries and paste into your report to use as a template when typing your report.

What to submit

Submit the following files:

reportA1.pdf

process.c

thread.c