Assignment 1

Operating Systems, Spring 2018

Time: One week

Instructions: Do not copy material from other sources, if it is necessary, then provide the references. Plagiarized assignment will get negative marks, and can be called for DC action.

You have to implement a parallel marge sort algorithm. For this you need to understand the merge sort algorithm. Please have a look at the Wikipedia page of merge sort `https://en.wikipedia.org/wiki/Merge_sort`.

The difference in your implementation is, at each level when you divide the array you `fork()` a child and delegate the responsibility to sort the sub-array to the child. You will use `socket()` to send the input array to children and receive the sorted array. Each parent only merges the sorted arrays, hence the process which initiated the whole procedure will merge two sorted arrays, returned from left sub-tree and right sub-tree. You should consider that you do not need to create a child when there are only two or less elements in the array. Based upon this rule the resulting tree will be complete but not necessarily full.

Additionally, there is a naming scheme for each process and the output file it creates. The parent process will have the name `P`, its child who handles left sub-tree will have the name `P.l`, while the child handling the right sub-tree will have the name `P.r`. Similarly, `P.l` will have children `P.l.l` and `P.l.r`, so on and so forth. The names matter a lot, because each process writes the sorted/unsorted sub-arrays into a file named after its own name (detail comes next).

So the parent will write a file named `P.txt` while its left child will write a file named `P.l.txt` and its right child will write a file named `P.r.txt`. All other processes which do sorting will also write a file named after their name. The file will contain the unsorted array they received as input and the sorted array below it. See the sample file for clarity. The sample input and output files are made for the example given in the wikipedia page for your ease. Carefully, look at them and follow the pattern.

Submission The submission should contain only three following things. If there is one missing then the assignment will be given 0 marks. Additional things will also be penalized.

1. `Assign-1.cpp` the code of the assignment
2. `InputAssign-1.txt` which contains the array to be sorted
3. `Makefile` which makes executing `make` command in the respective directory compile the code into an executable.

After executing `make` an executable should be created named `assign-1.o`. Executing that executable should generate all the respective output files.