DESIGN

- All filesystems are run on separate threads.
- A level by level traversal of the filesystems has been done and the list of common pathnames (including files and folders) is extracted. The level by level traversal is done using barrier synchronization. Shared memory is used to store the pathnames and their corresponding count. The count is updated every time a pathname is encountered for a level. The shared memory is freed after every level.
- The list of common pathnames files is checked for common majority. This check is done for all filesystems and for each common pathname file and is again achieved using barrier synchronization. First the file is checked for its size, then for its signature contents (using md5) and finally for contents. All these matches are done using shared memory. For a file if it is present in a filesystem its count is updated and the common majority files are collected for next level check.
- At this step list of common majority files have been created. The list of common pathnames directories are now traversed in bottom up fashion and checked for common majority. The directories which have exactly same subtree in k/2 filesystems are collected in this step. This step is also performed using barrier synchronization so that the k filesystems can be compared in parallel. A bitvector is maintained in shared memory for each directory if its contents are in common majority. Directories whose bitvectors reach the threshold value are declared as common.
- The list of common majority files and folders (in terms of relative pathname) are printed in the end along with the information whether the filesystems are exactly same or completely divergent.

I have had some discussions with Ram Karnani regarding the assignment.