

CS214 Assignment 2

Group members: Rebecca Rice, Yequ Sun

Use make to compile our program. make creates three executables: compressR_LOLS, compressR_worker_LOLS, compressT_LOLS. compressR_LOLS and compressT_LOLS require two arguments: the name of the file you want to open and the number of parts you would like the file to be split into.

make clean removes all three executables.

If the user gives too many or too few arguments, if the input file does not exist or cannot be accessed by the user, or if the specified number of parts is less than 1 or greater than the number of characters in the file, the program quits with an error.

compressR_LOLS is multiprocess, while compressT_LOLS is multithread. After checking the input file, the programs will spawn the specified number of threads/processes. The thread version keeps an array of each child id, while the process version forks the specified number of times, and, if the process is a child, calls execv to run compressR_worker_LOLS.

The child process first opens the input file, then checks to see if the output file already exists and returns if it does. If not, it performs the LOLS algorithm on its section of the input file, skipping over any non-alphabetical characters, and writing the result to the new file. The child then closes the input and output files, frees memory, and returns.

Meanwhile, the parent is waiting on the child processes. The thread version goes through a loop, waiting on each thread id in its array to join. The process version waits until there are no more children.