

Data Race Detection with Archer

As shown in the lectures, one can use the Thread Sanitizer and the Archer library together, to detect data races in OpenMP programs.

In this exercise, you can reproduce the data race example (`prime`) from the lecture, to get used to the tools. There is both a C and a FORTRAN version.

Exercise:

1. load the Archer module: `module load archer`
2. copy the example to one of your folders: either the C version
`cp $MODULE_ARCHER_BASEDIR/examples/prime.c .`
or the FORTRAN version
`cp $MODULE_ARCHER_BASEDIR/examples/prime.f90 .`
3. generate the OpenMP parallel version, by compiling it with the `-fopenmp` flag,
i.e. `gcc -fopenmp -o prime prime.c -lm`
or `gfortran -fopenmp -o prime prime.f90`
4. run the code for different numbers of threads, and look at the result - you should see different numbers, due to the data race.
5. compile the code with the thread sanitizer option (see the `archer` command or the lecture slides for details).
6. use `detect_dr` to generate a data race report
7. use `show_dr` to “highlight” the lines in the code where the race appears.
8. fix the code, and then rerun the last three steps - the data race should be gone.