

Assignment:

1. Add OpenMP directives to the Fibonacci (or Compute Pi) codes to parallelize them.
2. Use OpenMP tasks to parallelize the recursive part of the code.
3. Once the code is parallelized, generate the following plots:
 - a. Runtime as a function of the number of threads.
 - b. Runtime as a function of problem size (increase N in either code) for 2, 4 and 8 threads.

(the above is preliminary, and needs testing)

Compiling and Running:

See the ‘compile’ script for examples on how to compile the serial and parallelized codes. Once the parallelized code is compiled, make sure to run:

```
$ export OMP_NUM_THREADS=N
```

To set the number of OpenMP threads to N (some integer). Then the code can be run with

```
$ ./foo.exe
```

where ‘foo’ is the filename of the executable.

Hints: ...