Do the exercises below on mordor8.eecs.utk.edu. GNU UPC as well as some required libraries are installed in /mnt/scratch/sw. You will need to add /mnt/scratch/sw/upc-4.5.1.2/bin to your execution path variable and the following to your LD_LIBRARY_PATH environment variable:

```
/mnt/scratch/sw/mpfr-2.4.2/lib
/mnt/scratch/sw/gmp-4.3.2/lib
/mnt/scratch/sw/mpc-0.8.1/lib
```

1. Compile and run the Hello World example code using 4, 8, and 16 threads – e.g.,

```
upc -o hello_upc hello_upc.c
hello_upc -n 4
```

2. We can calculate $\pi$ by approximating the integral $\int_{0}^{1} \frac{4}{1+x^2} dx$. The code in pi_serial.c implements this method sequentially. Implement a parallel version by
   - parallelizing the loop (work distribution)
   - implementing the global sum (reduction operation)

Run your parallel version with 4, 8, and 16 threads and calculate the speedup over the sequential version.

3. Looks at the UPC implementation of the 2D heat conduction problem in heat_upc.c. Explain the use of the local scratch arrays and the memget function. In what two ways might use of the scratch arrays improve performance?

4. Read the paper


   a. Briefly describe the three hybrid MPI+UPC execution models that are described in this paper.

   b. Which of these models could help memory-constrained MPI codes scale to bigger problem sizes? Explain. (Unfortunately, this model is not yet implemented by vendor UPC compilers!)