# Introduction to OpenMP

Jeremy Iverson

College of Saint Benedict & Saint John's University

# background

- A standard *directive*-based shared memory programming API
- API consists of a set of compiler directives along with a library functions.
  - Compiler directives used for specifying concurrency and synchronization
  - Library functions used for setting and getting runtime parameters
- Intended to support environments where OpenMP compiler support is not available.

# execution model

- Uses the *fork* / *join* model of parallelism
  - Execution begins with a single thread (master thread), which creates a pool of threads upon encountering a parallel region

# compiler api

important directives

- `parallel`
- `for`
- `critical`
- `atomic`

important clauses

- `num_threads`
- `private`, `shared`, and `default`
- `reduction`

# library api

- Requires inclusion of `omp.h` header file.

- `omp_get_thread_num()`
- `omp_get_num_threads()`
- `omp_set_num_threads(int num_threads)`
- `omp_get_wtime()`