Multicore Computing

Weijie Zhao

08/30/2022

OpenMP

- Open Multi-Processing
- An API that supports multi-platform shared-memory multiprocessing programming in C, C++, and Fortran

OpenMP: Quick Start

```
for (int i = 0; i < N;++i){
        b[i] = a[i] + 1;
}
```

OpenMP: Quick Start

```
#pragma omp parallel for schedule(static) num_threads(8)
for (int i = 0; i < N;++i){
        b[i] = a[i] + 1;
}
```

- g++ test.cc –fopenmp -o test -O2

- brew install libomp
- clang++ test.cc -o test -O2 -Xpreprocessor -fopenmp -I/usr/local/include -L/usr/local/lib -lomp

OpenMP: Quick Start

```
int sum = 0;


for (int i = 0; i < N; ++i){
        sum += a[i];
}
```

OpenMP: Quick Start

```
int sum = 0;
#pragma omp parallel for schedule(static) default(shared) reduction(+:sum) num_threads(8)
for (int i = 0; i < N; ++i){
        sum += a[i];
}
```

OpenMP: Slow Start

- #include<omp.h>
- void omp_set_num_threads(int num_threads)
- int omp_get_num_threads()
- int omp_get_thread_num()

- #pragma omp atomic (update/read/write/capture)
- #pragma omp critical

Gauss-Seidel Smoother

- Solving PDE

- Parallel
- Synchronization
- Lock
- Communication