Lab Session 2

Submission deadline: Feb 15, 11:59pm

Please submit your lab results and code through CatCourses, including Makefile and a short report (up to one page). You may find the OpenMP tutorial from the Lawrence Livermore National Lab useful (https://computing.llnl.gov/tutorials/openMP/).

1. **OpenMP hello world**

The first lab task aims to tell you how to compile and run a simple OpenMP program.

Copy the following program into a file named "openmp_simple.c"

```
#include <iostream>
#include "omp.h"
int main() {
  #pragma omp parallel
 {
   std::cout << "Hello World\n"
 }
 return 0;
}
```

Compile the above program

```
g++ -fopenmp -o openmp_simple openmp_simple.c
```

Change the number of OpenMP threads by setting the OMP environment variable

```
Export OMP_NUM_THREADS=4
./openmp_simple
```

2. **Hello world extension**

Extend the program above to make it parallel where every thread prints out its id and total number of threads.

Hints: using APIs, omp_get_num_threads() and omp_get_thread_num();

3. **Parallelization of matrix vector multiplication**

3.1 Create a program that computes a simple matrix vector multiplication b=Ax, Use OpenMP directives to make it run in parallel. Change the number of threads and measure performance.

3.2 Fix the number of threads (e.g., using 4 threads), and try static and dynamic scheduling strategies ("dynamic" and "guided"). Measure performance and explain your results.