Parallel and Distributed Programming
Laboratory 3

Steps:

1. Create a working directory (eg. lab3).
2. On the basis of the sample program test the various data-sharing clauses. Follow instructions in the provided code and try to understand behavior of each clause.
3. Write a program which will include four loops of 15 iterations for testing different strategies of sharing the work:
   a) static, chunk_size = 3
   b) static, default chunk size
   c) dynamic, chunk_size = 3
   d) dynamic, default chunk size

   A sample code:

```
threads = .... ;
omp_set_num_threads(threads);
chunk_size = .... ;
size = .... ;
#pragma omp parallel for schedule(...,chunk_size)
for(i=0;i<size;i++)
{
        a*=0.5;
        printf("Thread %d, index %d\n",omp_get_thread_num(),i);
}
```

4. Run the program and obtain the corresponding output for four threads.
5. Change the number of iterations to 150000, disable the data display in a loop and measure the execution time of each loop (with omp_get_wtime() function) – create a chart(in any program you like) and compare the scheduling options in terms of execution time.

To use OpenMP, you must enable it:

- In gcc/g++/linux compilers - add -fopenmp during compilation:
  gcc -fopenmp program.c -o program
- In Visual Studio, you must open Project Properties->Languages->C/C++->and search for Enable OpenMP option (/openmp)