Hands-on “OpenMP with Intel Xeon and Intel Xeon Phi Architecture”

Task 1 Compiling running and environment variable

The file hello_omp.c implements an example of openMP4:

1.1 Compile hello_omp.c to Intel Xeon

1.2 Compile hello_omp.c to Intel Xeon Phi

1.3 Execute the code on Intel Xeon with 16 threads (use env var OMP_NUM_THREADS)

1.4 Execute the code on Intel Xeon Phi with 100 threads (use env var OMP_NUM_THREADS)

Task 2 Thread affinity (use env var OMP_NUM_THREADS and KMP_AFFINITY)

2.1 Execute hello_omp with 10 threads and using affinity policy to allocate threads close to each other (compact)

2.2 Execute hello_omp with 10 threads and using affinity policy to spread threads among processors (scatter)

2.3 Execute hello_omp with 10 threads and using affinity policy to spread threads among processors on Xeon Phi (scatter)

2.4 Execute hello_omp with 10 threads and using affinity policy to balance the thread allocation among processors on Xeon Phi (balanced)

Task 3 data transfer between host and devices using OpenMP4:

Compile the code omp3.c

Turn the offload report on

Run omp3

Note that the in first call of function test the offload was not performed and the code was executed at the host, because of clause if.

What variables were transferred to and from MIC 2?

Task 4 Implementation of several offload operations between host and devices using OpenMP4

Include in the code omp4.c (after line 28) the following sequence of commands:

- Copy variables A, B and C to MIC 0 using omp update
- Execute sum2 with value A, B and C on MIC 0
- Copy variable sum from MIC 0 using omp update
- Copy variable sum to MIC 1 using omp update
- Execute multiply2 with value sum on MIC 1
- Copy variable mult from MIC 1 using omp update

Task 5 Vectorization of functions:

Compile the file OMP4-7.c with compilation report:

```
icc OMP4-7.c -o OMP4-7 -fopenmp -vec-report6

cp OMP4-7.optrpt OMP4-7.optrpt2
```

Include **pragma omp declare simd** in top of functions **min** and **distsq** and compile OMP4-7.c again.

```
icc OMP4-7.c -o OMP4-7 -fopenmp -vec-report6
```

Compare the compilation report of both compilations, and note that the functions were vectorized.

```
diff OMP4-7.optrpt OMP4-7.optrpt2
```