

Hands-on: Checking thread level in MPI

- Choose either C/C++ (`c`) or Fortran (`fortran`) samples. Both of them are fine, as well.

C/C++

How to compile and how to execute

1. Compile program

- The executable file (`chk_thd_level`) is generated in `c/`.

```
$ cd c
$ make # make -f Makefile.own # if using own compiler
$ ls
chk_thd_level ...
```

2. Run program

- You can run the program either:

```
## To run as a batch job
$ cd c/results
$ pjsub task.sh
## Or, to run in an interactive job
$ cd c/results
$ bash task.sh
```

- The job in the Exercises will be completed within 1 minutes.
 - For safety, we set the elapsed time of the job scripts as 3 minutes.

Exercises A

- E1: Examine the meaning of each thread level in MPI (`MPI_THREAD_SINGLE`, `MPI_THREAD_FUNNELED`, `MPI_THREAD_SERIALIZED`, and `MPI_THREAD_MULTIPLE`).
- E2: Check the result of `chk_thd_level`. What kinds of thread level is acceptable for Fujitsu MPI?

Exercises B (advanced)

- E3: Intel MPI benchmark also provide a similar program of checking the thread level. Try `use_imb/` using your built IMB executable file.

Fortran

How to compile and how to execute

1. Compile program

- The executable file (`chk_thd_level`) is generated in `fortran/`.

```
$ cd fortran
$ make # make -f Makefile.own # if using own compiler
$ ls
chk_thd_level ...
```

2. Run program

- You can run the program either:

```
# To run as a batch job
$ cd fortran/results
$ pjsub task.sh
# Or, to run in an interactive job
$ cd fortran/results
$ bash task.sh
```

- The job in the Exercises will be completed within 1 minutes.
 - For safety, we set the elapsed time of the job scripts as 3 minutes.

Exercises A

- E1: Examine the meaning of each thread level in MPI (`MPI_THREAD_SINGLE`, `MPI_THREAD_FUNNELED`, `MPI_THREAD_SERIALIZED`, and `MPI_THREAD_MULTIPLE`).
- E2: Check the result of `chk_thd_level`. What kinds of thread level is acceptable for Fujitsu MPI?

Exercises B (advanced)

- E3: Intel MPI benchmark also provide a similar program of checking the thread level. Try `use_imb/` using your built IMB executable file.