BE OpenMP

03/03/2022

This BE is made of five different exercises of different difficulty for a total of 100 points:

Part 1 `norm1`: an exercise on the parallelization of the norm1 computation of a matrix. **20 points**.

Part 2 `ring`: an exercise about processing a token sequential in a ring of threads. **20 points**.

Part 3 `sched`: an exercise on the parallelization of loops with different schedulings. **15 points**.

Part 4 `server`: an exercise on the parallelization of a server that processes requests. **20 points**.

Part 5 `sparse_nn`: an exercise on the parallelization of a sparse neural network. **25 points**.

For each exercise, detailed explanations and instructions are given in the `subject.pdf` file inside the corresponding directory.
All the exercises include some coding tasks: these tasks consist in writing, compiling and executing some OpenMP parallel code and are identified by the keyboard symbol .
Some exercises also include more theoretical questions that are identified by the pencil symbol .
If you want to provide details or explanations about your work in textual form, you can do it in the form of comments directly in the source code.

General advice:

- When implementing the parallelization, test your code on small data. When you're sure everything works fine, increase the size of data to evaluate performance.
- All the proposed parallel solutions have to work with any (reasonable) number of threads. This means that the parallel code has to work also in

the case where only one thread is used. Check your parallel code with one thread first; this case will be easier to debug in case of problems. Then test with more threads.

- The amount of coding required in each exercise is relatively small. If you find yourself writing a lot of code, you're probably on the wrong track.
- OpenMP tasks is a very powerful tool but also usually introduces some overhead. Use this feature with care.

Important

Once you have finished execute the `pack.sh` script like this:

```
$ ./pack.sh
```

This will generate a package containing the code you have developed and the responses you have provided. This file is named

`username.tgz`

where `username` is your username.

Upload this file into the OpenMP course page on Moodle in the section corresponding to your room.

Before leaving verify with the supervisor in your room that the package has been received.