Documentation for sample code

Adrián Lozano-Durán and Guillem Borrell

December 4, 2015

1 Folder information

This package contains the next folders:

1. *Src*: source files for the sample codes. Serial and Coarrays.
2. *Doc*: documentation (this file).
3. *data*: input data required to execute the example.

2 Compilation

To compile the code with GNU Fortran go to the *Src/fortran_serial/* folder and run in a linux terminal:

```
make genus
```

To compile with Intel® Fortran Compiler edit the *F90* variable in the *Makefile* and use *F90=ifort*. Modify the flags *F90FLAGS* and *LFLAGS* accordingly. To compile the Fortran Coarrays version go to *Src/fortran_coarrray/* and run `make genus_coarray`.

3 Usage

To run the code, in a linux terminal:

```
./genus <file-name>
```

where file-name contains the raw data.

4 Example

To run the example:

```
./genus ../../data/random.128.dat
```

The results should match those from case *Random2* in Ref. [1], table III.

For the Fortran Coarrays version with OpenCoarray - GNU Fortran[1]

[1] IMPORTANT NOTE: By the time this document was generated, the *critical constructs* cause an internal compiler error with OpenCoarray version 1.0.3. In that case, comment all the *critical constructs* and the code should work. Note that this change will not affect the output of the program, but could lead to I/O bottlenecks when the number of images used is large. Intel Fortran Compiler at version 2016 16.0.0 20150815 is free from this issue.

```
mpiexec -n <N> ./genus_coarrays ../../data/random.128.dat
```

where N is the number of processing images.

For the Fortran Coarrays version with Intel® Fortran Compiler

```
./genus_coarrays ../../data/random.128.dat
```

Use the environment variable *FOR_COARRAY_NUM_IMAGES* to set the number of processing images.

5 References

[1] Adrián Lozano-Durán and Guillem Borrell '*An efficient algorithm to compute the genus of discrete surfaces and applications to turbulent flows*' ACM TOMS, 2015.