# MPI: Practical 1 Solutions

## 1 My First MPI Program

You should find that you get:

```
Hello parallel world, I am process <n> out of 8
```

for each process `<n>`.
The lines may not be displayed in ascending order; this is not a bug in MPI or the program.

## 2 Using different MPI implementations

The CSC network has both OpenMPI and MPICH installed.

The only difference with MPICH is that it seems to allow output of partial lines, resulting in:

```
Hello parallel world, I am process Hello parallel world, I am process Hello parallel world,
I am process 1 out of 6
3 out of 6
4 out of 6
```

Again, this is not a bug in MPICH.

## 3 Mixing different MPI implementations

Running an OpenMPI executable with mpiexec.mpich (or vice versa) gives:

```
Hello parallel world, I am process 0 out of 1
```

for **all** processes.

## 4 MPI Wrappers

OpenMPI's wrapping gives:

```
g++ hello_world.cpp -o hello_world -I/usr/lib/x86_64-linux-gnu/openmpi/include/openmpi
-I/usr/lib/x86_64-linux-gnu/openmpi/include -pthread -L/usr/lib/x86_64-linux-gnu/openmpi/lib
-lmpi_cxx -lmpi
```

MPICH's wrapping gives:

```
g++ -Wl,-Bsymbolic-functions -Wl,-z,relro hello_world.cpp -o hello_world
-I/usr/include/x86_64-linux-gnu/mpich -L/usr/lib/x86_64-linux-gnu -lmpichcxx -lmpich
```

## 5 MPI Version

On the CSC machines, the default MPI version supported by both OpenMPI and MPICH is 3.1

# 6 Parametric studies

The example given is for OpenMPI. The equivalent for MPICH would be:

```
$ mpiexec.mpich -n 4 /bin/bash -c 'echo I am rank $PMI_RANK of $PMI_SIZE'
```

For parametric studies, this only makes efficient use of the processors if all jobs take approximately the same time to run.