

MPI: Practical session 6

1 Short Receives

Write a master-slave program that computes the number of primes in sets of equal ranges. The range size should be sufficiently small that each processor does approximately the same amount of work.

(To determine whether a number is prime, do not try anything fancy; just check all possible odd divisors less than \sqrt{N} .)

Use `MPI_Get_count` to determine the number of primes that each slave has computed in each range.

2 Deadlock

Compile and run the `deadlock.c` code (slide 176) on two processes.

Now try varying the value of `SIZE` and retrying the same run. For what values of `SIZE` does the code (slide 176) complete? If it fails to complete, does it sit using no CPU time, or using CPU time?

(To find the exact value of `SIZE` for which the code deadlocks, try using a bisection method.)

3 Ping-pong example

Download the `pingpong.c` example, compile, and run it on two processors. See what bandwidth you can get between two cores.