

Informe práctica I Paralelismo

En esta práctica se implementó a partir de un programa en C, que calculaba el número de primos menores que un valor introducido por teclado, un SPMD usando la librería de MPI (mpi.h).

Al principio del programa declaramos las variables necesarias para el funcionamiento, después inicializamos los atributos MPI en el canal `MPI_COMM_WORLD`, donde *numprocs* es la cantidad de procesos y *rank* el número de cada proceso.

Después entramos en un bucle infinito (*while(1)*) que siempre repetirá la ejecución del programa hasta que se introduzca un cero, en ese caso hará un *break* para salir. En el caso de que no sea un 0, cogemos nuestro número de primos con el proceso 0, es decir, sólo pedirá el número un proceso, en nuestro caso el 0. Después de solicitar el número por pantalla este se envía a todos los demás procesos con un bucle *for* que va incrementando la variable *i*, hasta que sea igual al número de procesos, y un *MPI_Send* que va enviando el dato a cada proceso según el valor de *i*. El resultado del envío se comprueba con *MPI_SUCCESS*.

Mientras, el resto de procesos están esperando a recibir el dato (con *MPI_Recv*), aquí también comprobamos que se haya recibido el dato correctamente. Una vez el proceso '*i*' recibe el dato, tiene que calcular los primos que le corresponden, si hay *n* procesos significará que cada uno tendrá que hacer x/n operaciones siendo *x* el número introducido por pantalla. Esto se consigue sumando a la variable '*i*' del bucle *for* el número de procesos en cada iteración. Así para 3 procesos, el cero empezará en el 2, el uno empezará por el 3 y el dos por el 4, y en la siguiente iteración serían 5, 6, 7, y así hasta llegar al número *x*. Cada proceso guardará individualmente la suma de los primos.

Una vez hechos los cálculos, cada proceso enviará sus resultados al principal (el proceso 0) con un *MPI_Send*. Este recibirá con la función *MPI_Recv* y un bucle *for* el resultado de la cuenta de primos de cada nodo, este los irá sumando a la suya propia (el proceso 0 también hace una cuenta parcial como el resto) y después sacará el resultado por pantalla.