



Universidad de Costa Rica
Facultad de Ingeniería
Escuela de Computación e Informática

Arquitectura de Computadoras
CI-1323
Grupo 001
Tarea Programada con MPI

Profesora: Ileana Alpizar
Estudiante: Georgina Santamaría B05954

Martes 30 de Agosto, 2016

Prueba inicial:

Prueba realizada con 2 procesos y una matriz de tamaño $n=10$, para corroborar que estuviera bien los cuatro puntos descritos en la tarea, contar primos, contar primos por filas, multiplicación de la matriz por un vector, crear la matriz B y tiempo de ejecución. A continuación el pantallazo de esta prueba:

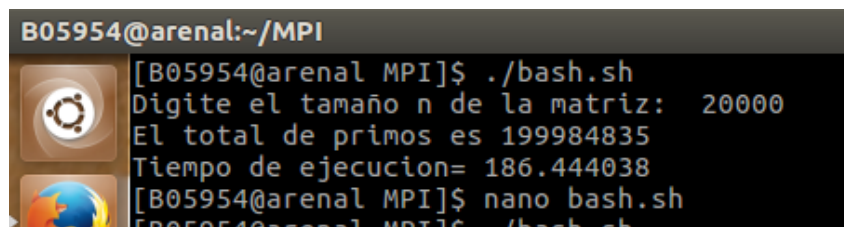


```
[B05954@arenal MPI]$ ./bash.sh
Digite el tamaño n de la matriz: 10
La matriz M es:
004 009 001 004 005 002 001 007 002 001
003 004 005 006 006 008 002 009 006 003
004 006 001 000 004 008 001 001 006 007
008 003 007 002 004 004 004 008 007 008
002 006 000 000 008 001 009 003 007 003
008 005 005 006 008 008 004 007 001 004
000 006 007 005 003 007 002 001 009 006
001 009 007 003 008 003 000 003 004 008
002 006 001 002 000 009 005 005 004 006
003 001 006 001 005 007 002 003 008 004
=====
El total de primos es 51
Primos por fila:
#0      cantidad:7
#1      cantidad:4
#2      cantidad:4
#3      cantidad:4
#4      cantidad:5
#5      cantidad:4
#6      cantidad:6
#7      cantidad:5
#8      cantidad:5
#9      cantidad:7
=====
El vector V es:
0,5,0,2,1,3,9,1,3,2,
=====
El vector Q es:
88,113,100,116,152,123,122,99,135,86,
=====
La matriz B es:
016 018 019 016 017 016 012 019 016 006
015 027 017 021 029 026 021 025 026 017
021 018 019 013 022 025 016 025 027 024
017 030 013 013 022 021 026 023 036 025
024 016 018 016 021 030 021 034 021 022
015 030 023 024 033 028 030 016 028 014
015 027 030 024 031 023 014 022 021 027
012 029 027 025 017 027 013 013 028 024
012 019 022 007 024 024 021 020 027 022
006 016 009 014 013 023 017 018 019 018
=====
Tiempo de ejecucion= 0.000052
[B05954@arenal MPI]$
```

LOS ARCHIVOS .TXT SOLICITADOS EN EL ENUNCIADO JUNTO CON EL CODIGO SE ENCUENTRA EN LA DIRECCIÓN HOME/B05954/MPI

Pantallazo con Caso A:

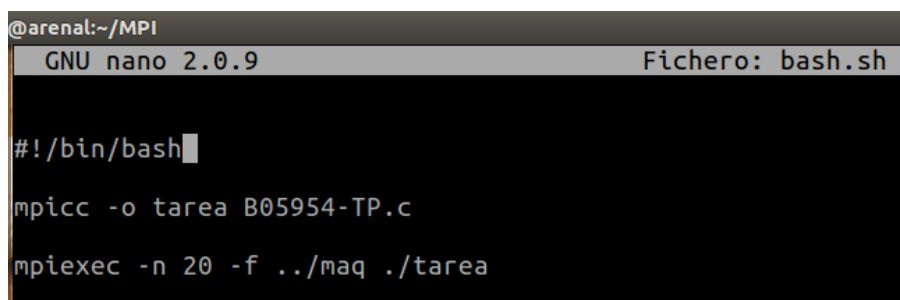
Tuve que quitar los printf (para imprimir en pantalla) para imprimir el resultado de matrices, solamente dejé el total de primos y el tiempo de ejecución para que lo mostrara en consola, ya que duraba mucho en dar resultados.



```
B05954@arenal:~/MPI
[B05954@arenal MPI]$ ./bash.sh
Digite el tamaño n de la matriz: 20000
El total de primos es 199984835
Tiempo de ejecucion= 186.444038
[B05954@arenal MPI]$ nano bash.sh
[B05954@arenal MPI]$ ./bash.sh
```

Pantallazo Caso B:

Tuve que quitar los printf (para imprimir en pantalla) para imprimir el resultado de matrices, solamente dejé el total de primos y el tiempo de ejecución para que lo mostrara, ya que duraba mucho para dar resultados, incluso mas de una hora y no tiraba los resultados corriendo).

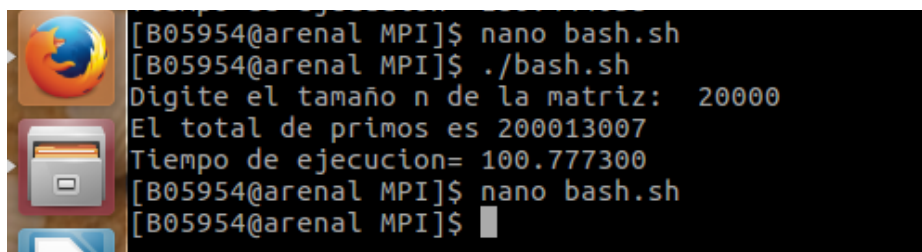


```
@arenal:~/MPI
GNU nano 2.0.9                                Fichero: bash.sh

#!/bin/bash

mpicc -o tarea B05954-TP.c

mpiexec -n 20 -f ../maq ./tarea
```



```
[B05954@arenal MPI]$ nano bash.sh
[B05954@arenal MPI]$ ./bash.sh
Digite el tamaño n de la matriz: 20000
El total de primos es 200013007
Tiempo de ejecucion= 100.777300
[B05954@arenal MPI]$ nano bash.sh
[B05954@arenal MPI]$
```

Resultados	Caso a 1proceso , n=20000	Caso b 20procesos , n=20000
Tiempo total	186.444038 segundos	100.777300 segundos
Tiempo sin despliegues en pantalla ni escrituras de archivos	186.444038 segundos	186.444038 segundos

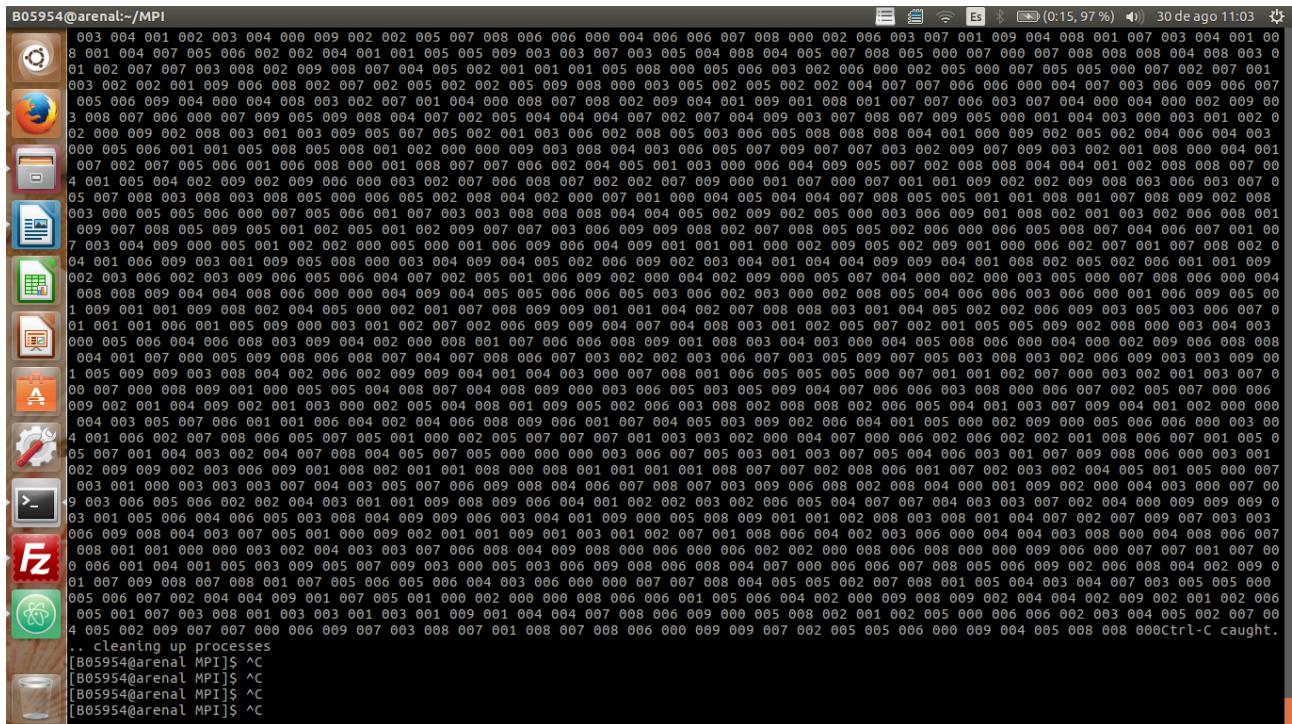
A como podemos ver los resultados entre más procesos se tienen menos tiempo de ejecución. Como por ejemplo con 1 solo proceso dura más que con 20 procesos, la diferencia es de 86 segundos.

Ambos resultados dan lo mismo , tanto para el tiempo total como el tiempo sin despliegues, debido a que el tiempo lo calculé antes que comenzará a desplegar resultados. El paso de mensajes mediante el uso de MPI, permite que una tarea la cual puede tardar en realizarse una cantidad de operaciones repetitivas, se pueda minimizar este tiempo al dividir las operaciones a diferentes procesos, y el resultado final se lo maneje un proceso principal para mostrar los datos completados.

Problemas no resueltos y posibles soluciones:

El programa corre bien y realiza correctamente las 4 cosas solicitadas. El único problema que no se logró , fue que despejara la matriz en consola junto con los resultados para los casos de prueba ya que teníamos un n muy grande y duraba demasiado, incluso llegué hasta

esperar 1:20min , si desplegaba y podemos verlo en la siguiente imagen:

A screenshot of a terminal window titled 'B05954@arenal:~/MPI'. The terminal displays a continuous stream of binary data (0s and 1s) across multiple lines. At the bottom, the prompt 'B05954@arenal MPI]' is visible, followed by several lines of output: '...', 'cleaning up processes', and 'Ctrl-C caught.' The terminal window has a standard Linux desktop environment background with various icons on the left side.

pero si se dejaba corriendo iba a despegarlo correctamente, ya que con n pequeños lo hacia, la prueba está en la prueba primera que realicé con $n=10$ y 2 procesos. Se hubiese podido resolver si hubiera tenido un poco mas de tiempo para dejar corriendo el programa y sin que el cluster se nos llene tanto.