

**Universidad de Costa Rica
Escuela de Ciencias de la Computación e Informática
CI-1323 Arquitectura de Computadoras**

Tarea Programada 10%

Grupo 01

Prof. Ileana Alpízar

Elaborado por:

Carlos Esquivel Sinfonte
Daniela Quesada Aguilar

A62066
B25247

I Semestre 2018

Codigo Fuente



TP.c

Imágenes de Resultados obtenidos

Caso a:

```
[A62066@arenal TareaProgramadaMPI_CarlosyDaniela]$ mpiexec -n 1 ./tar_mpi  
Por favor digite el tamaño de la matriz cuadrada 2000
```

```
*** RESULTADOS FINALES ***
```

```
Valor de n = 2000
```

```
Cantidad de procesos que corrieron = 1
```

```
Total de primos en M (tp) = 1598629
```

```
Duración del programa sin entrada o salida de datos = 0.446407
```

```
Duración TOTAL del programa = 3.464187
```

Caso b:

```
[A62066@arenal ~]$ mpiexec -n 20 ./tar_mpi
Por favor digite el tamaño de la matriz cuadrada 2000

*** RESULTADOS FINALES ***

Valor de n = 2000
Cantidad de procesos que corrieron = 20
Total de primos en M (tp) = 1598683

Duración del programa sin entrada o salida de datos = 3.538919

Duración TOTAL del programa = 11.525443
```

Caso c:

```
[A62066@arenal ~]$ mpiexec -n 10 ./tar_mpi
Por favor digite el tamaño de la matriz cuadrada 30

*** RESULTADOS FINALES ***

Valor de n = 30
Cantidad de procesos que corrieron = 10
Total de primos en M (tp) = 391

Matriz M = Matriz 30 x 30
8 0 7 8 9 8 2 2 4 1 5 7 8 5 6 5 9 0 8 7 7 5 4 0 3 5 6 3 9 3
7 5 2 5 6 3 7 8 9 8 5 8 8 3 4 3 2 6 2 9 6 9 4 6 4 1 4 9 0 7
7 4 1 4 0 4 2 1 3 2 8 1 1 1 8 6 6 6 0 7 4 9 3 0 3 5 4 2 5 3
5 7 2 9 0 6 3 1 1 6 1 4 9 3 4 5 1 3 6 8 7 7 2 8 3 9 4 7 6 8
1 7 0 3 7 6 6 0 9 2 2 6 1 7 0 9 8 3 5 6 1 4 0 1 7 9 5 5 6 3
7 3 4 2 0 2 9 3 2 3 5 0 7 2 0 6 2 6 3 8 9 7 9 0 5 8 7 0 4 3
4 6 7 6 8 2 7 1 6 1 8 6 8 2 8 6 4 4 9 3 3 6 4 5 1 2 5 1 9 8
5 4 6 3 2 8 1 6 4 4 4 2 4 9 1 0 3 5 9 6 1 5 2 8 9 5 3 0 6 1
1 7 6 4 2 4 7 0 1 3 4 3 7 5 4 9 9 2 8 7 5 6 0 3 5 7 9 7 3 2
4 0 5 0 4 1 0 4 2 3 0 8 3 6 4 2 5 6 3 4 3 1 4 7 8 1 8 7 4 0
0 2 1 3 8 5 5 2 7 0 5 7 3 1 2 8 8 8 3 2 2 6 9 1 4 0 1 3 7 3
9 6 3 4 4 8 9 9 8 0 4 5 5 7 7 3 5 8 8 9 2 4 8 5 0 1 6 4 0 9
1 7 1 6 4 0 7 5 0 7 9 5 5 6 5 0 3 5 9 2 7 3 1 2 4 4 0 0 4 1
5 1 7 3 5 9 1 0 1 0 1 7 5 7 4 6 0 9 7 4 8 0 7 2 6 1 2 8 7 3
5 6 4 8 1 5 1 3 7 2 4 6 9 3 2 7 5 1 4 0 1 7 7 3 5 1 5 4 8 0
3 7 6 4 0 4 5 4 1 7 0 9 9 4 1 6 9 4 1 1 5 8 9 0 3 0 5 7 0 7
4 6 8 8 7 2 5 3 5 5 5 4 4 9 7 0 1 3 3 4 8 1 5 0 5 5 5 4 8 5
4 3 5 4 0 7 9 5 3 4 1 9 0 2 7 1 5 0 4 0 1 8 5 1 5 3 9 1 3 6
7 9 3 1 2 2 0 9 5 7 0 7 7 5 4 0 2 6 7 4 9 5 2 3 0 7 4 1 0 2
7 1 4 0 3 7 0 3 4 9 5 1 7 2 7 9 7 6 6 1 5 1 5 0 3 2 6 5 2 8
1 0 3 1 5 0 4 0 4 3 8 8 2 2 5 2 9 1 8 3 9 2 8 1 5 1 8 0 5 6
9 8 2 2 6 4 6 8 0 2 6 8 4 9 6 8 0 7 9 3 8 1 4 0 6 5 0 9 2 8
3 7 2 6 5 8 2 3 8 6 0 7 1 1 5 1 3 2 0 7 3 1 1 5 0 8 5 9 2 3
6 2 3 2 7 1 6 3 1 3 3 0 6 6 5 7 9 0 0 8 3 3 9 8 3 9 5 4 2 0
0 5 0 2 2 1 8 7 3 3 0 3 9 8 1 9 0 1 9 8 4 2 0 4 8 9 1 2 3 1
3 6 8 7 1 2 8 8 7 1 1 2 1 1 3 0 1 4 1 2 0 3 2 6 1 3 7 6 9 1
2 7 6 0 0 0 8 1 7 1 4 2 2 5 5 4 9 2 5 1 2 1 5 9 4 8 8 2 7 9
2 9 5 4 1 7 4 2 5 5 8 7 0 4 4 4 5 6 2 6 1 5 1 2 1 1 0 4 9 2
3 1 9 7 4 9 1 7 4 8 5 3 8 1 9 5 6 8 9 4 9 4 5 3 5 9 5 9 0 7
3 0 5 0 5 4 3 4 0 3 1 5 2 5 5 9 1 3 8 6 4 2 3 9 9 4 4 1 4 4

Vector V = Matriz NO CUADRADA 1 x 30
2 0 1 1 3 3 3 2 4 4 1 0 1 1 1 0 0 1 3 4 1 4 0 4 0 3 1 3 4 4

Vector Q (Q = M*V) = Matriz NO CUADRADA 1 x 30
293 349 208 323 262 236 300 269 253 183 206 324 204 217 209 199 261 219 232 231 191 259 261 2
32 229 233 249 233 324 230

Vector P (primos por columna) = Matriz NO CUADRADA 1 x 30
18 14 14 9 11 9 13 14 15 16 13 14 13 16 14 9 13 10 10 10 15 16 12 12 17 16 13 12 8 15
```

Matriz B = Matriz 30 x 30

```
15 20 17 29 31 22 19 16 16 18 18 28 28 22 20 23 16 23 17 31 25 25 13 13 12 15 18 27 15 19
27 18 20 25 23 28 22 27 32 25 34 29 28 21 24 20 26 16 25 31 35 33 26 14 17 19 24 18 30 13
23 24 13 19 14 15 17 15 16 27 17 22 20 16 23 28 21 21 21 28 33 32 18 20 15 22 19 27 16 23
20 25 19 18 22 19 18 6 20 12 21 21 18 24 20 25 23 19 22 34 27 29 20 14 30 30 29 24 32 20
20 18 16 21 16 27 24 19 14 22 16 13 30 13 20 28 23 25 23 28 27 19 16 16 25 38 30 23 24 20
15 27 16 15 19 19 27 15 23 13 18 24 18 18 16 23 26 18 31 29 28 35 20 20 21 31 25 17 22 18
22 24 29 26 18 27 20 23 14 22 24 24 27 29 17 24 19 28 28 29 22 25 26 18 22 21 18 15 28 21
14 28 26 21 23 17 29 12 21 16 22 19 30 21 22 19 21 23 37 26 20 20 19 27 28 26 22 17 19 17
17 18 28 15 16 22 12 18 10 15 14 24 22 31 23 24 28 30 29 30 22 17 15 23 32 27 34 26 22 6
5 18 12 16 15 14 17 8 17 8 20 21 27 19 18 28 30 24 24 19 15 20 21 23 25 24 26 29 21 9
15 9 14 16 24 27 21 27 19 15 16 28 19 19 22 23 34 33 24 20 15 22 28 26 13 7 18 22 17 19
16 27 15 20 28 26 38 33 24 19 23 26 25 26 24 23 27 34 37 23 24 23 27 16 14 11 12 13 24 13
22 16 24 18 19 28 22 21 21 16 26 31 26 30 22 17 13 34 31 31 22 15 21 14 16 10 12 16 12 17
12 26 16 29 22 20 18 10 8 11 21 24 33 25 24 17 23 22 33 21 20 25 17 20 18 14 16 21 30 11
19 23 31 20 19 20 15 15 14 20 13 35 32 25 17 26 22 23 13 10 21 23 33 17 18 12 17 32 19 18
19 28 29 26 16 16 19 16 24 15 25 28 35 26 20 23 25 18 13 11 23 30 29 15 13 14 22 20 30 12
17 28 33 31 17 25 24 22 17 26 15 31 26 26 24 15 18 11 15 16 19 30 20 11 18 18 28 25 20 26
18 27 23 18 20 20 26 29 22 20 19 21 22 23 21 13 9 18 14 13 26 20 21 14 14 29 22 18 18 16
27 23 22 10 8 18 20 22 28 25 20 24 26 20 23 16 20 21 27 21 24 25 20 6 18 16 27 11 8 16
16 21 11 9 17 12 14 16 25 28 23 28 19 23 27 25 33 26 28 19 25 18 16 12 10 19 25 14 20 18
17 13 10 11 15 20 10 19 11 26 30 27 23 20 22 33 19 31 27 24 27 21 20 14 16 21 15 27 15 27
21 26 17 17 22 24 24 17 22 17 24 33 24 22 33 17 27 19 27 30 24 16 14 16 16 20 27 20 26 19
25 22 20 17 32 20 25 24 18 19 22 16 19 22 18 24 15 12 18 21 22 9 20 14 22 27 27 29 18 13
11 23 9 20 17 23 20 20 18 16 6 19 22 26 24 31 19 12 17 26 21 18 21 29 28 34 24 22 11 6
14 13 18 13 13 14 30 29 21 10 10 14 27 25 26 17 20 14 19 31 17 12 17 26 25 30 24 16 17 5
11 29 27 18 12 12 34 31 26 13 8 9 15 18 10 17 14 9 21 12 11 8 16 22 22 28 25 26 26 20
14 30 26 17 2 17 21 26 21 18 16 17 10 17 21 22 21 26 11 16 5 16 18 26 23 24 25 27 36 19
16 24 33 17 16 21 22 19 23 27 29 20 21 14 26 22 30 23 28 14 23 12 18 16 13 19 18 24 22 27
9 22 27 24 26 25 24 18 24 25 25 28 14 27 24 33 25 32 31 34 22 25 16 24 27 24 27 19 29 13
6 9 14 17 13 21 12 14 11 12 14 11 20 13 28 20 19 20 26 22 21 13 19 24 27 26 14 18 9 15
```

Duración del programa sin entrada o salida de datos = 0.041770

Duración TOTAL del programa = 1.197260

Tabla de resultados de los casos

Resultados	Caso a	Caso b	Caso c
	1 proceso, n = 2000	20 procesos, n = 2000	10 procesos, n = 30
Tiempo total	3.464187	11.525443	1.197260
Tiempo sin despliegues en pantalla ni escrituras de archivos	0.446407	3.538919	0.041770

Analisis de Resultados

Como se puede observar, el hecho de tener mas procesos corriendo en paralelo no mejora el tiempo de ejecucion, sino que mas bien lo empeora. Esto puede deberse al overhead que existe al tener que consolidar toda la informacion, ademas de la limitante en cuanto a recursos del cluster arenal. Hay que tomar en cuenta tambien que la cantidad de trafico por parte de otros estudiantes podria impactar o "ensuciar" la muestra.

Habiendo dicho lo anterior, una matriz cuadrada de $n = 2000$ podria no representar un problema suficientemente grande como para que la paralelizacion tenga un efecto positivo en el tiempo de ejecucion del programa.

Solo por probar, se realizo una prueba con una matriz de 10000 y parece apoyar la teoria de que mientras mas crece la matriz, la velocidad de procesamiento disminuye cuando se ejecuta el problema con mas procesos.

Resultados	Caso y 1 proceso, n = 10000	Caso z 10 procesos, n = 10000
Tiempo sin despliegues en pantalla ni escrituras de archivos	11.063989	21.319378

Se puede ver que hay una diferencia de $21.319178/11.063989 = 1.926898$ y entre la primera muestra $3.538919/0.446407 = 7.927562$