

# INSTITUTO POLITECNICO NACIONAL

ESCUELA SUPERIOR DE COMPUTO

---

## TAREA 3 Multiplicacion distribuida de matrices utilizando pasos de mensajes

Desarrollo de Sistemas Distribuidos

---

*Nombre:* Morales Garcia Christian Arturo

*Grupo:* 4CV2

*Fecha de entrega:* mar 12 2020

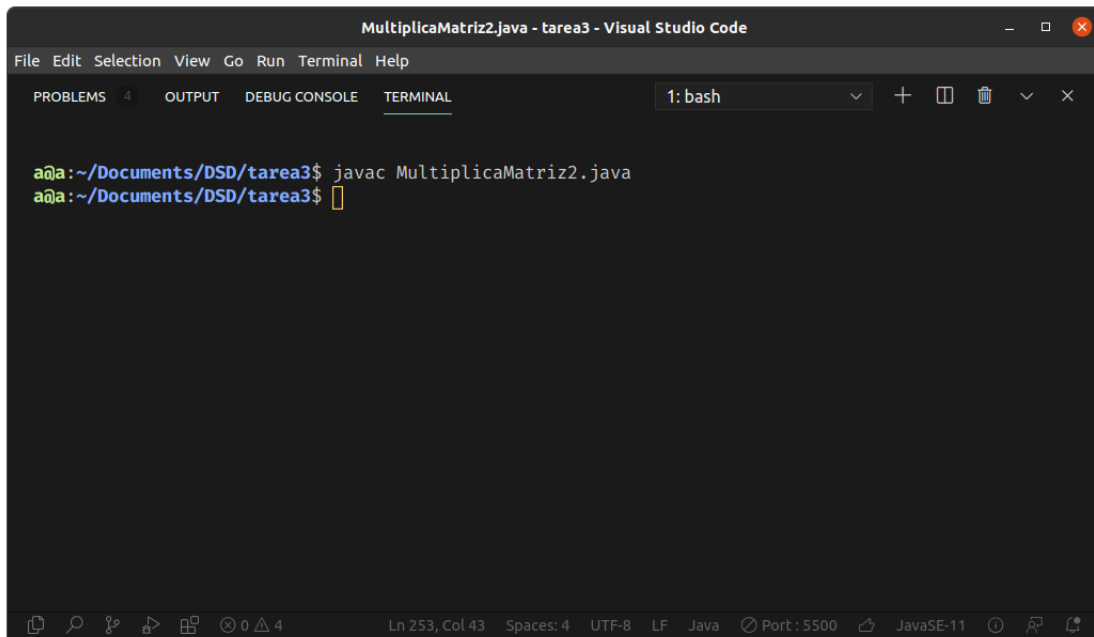
*Fecha:* Marzo 12 2020

# 1 Compilación

Para compilar usamos el siguiente comando:

```
javac <nombre_programa>.java
```

Podemos observar que se ha compilado sin ningún problema.



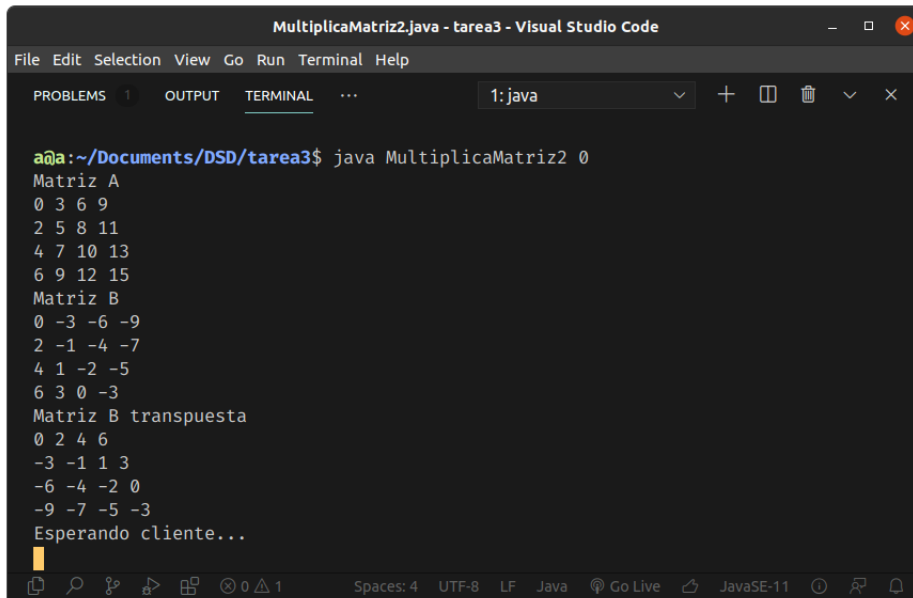
# 2 Ejecución de programa

El comando para ejecutar el programa es:

```
java <nombre_programa_compilado> <numero_de_nodo>
```

## 2.1 Inicio de ejecución de servidor

Recordemos que para iniciar el programa primero debemos iniciar el servidor y el servidor es el nodo cero por lo que es necesario colocar el "0" después de nombre del programa compilado. En la siguiente imagen se muestra el comando completo:



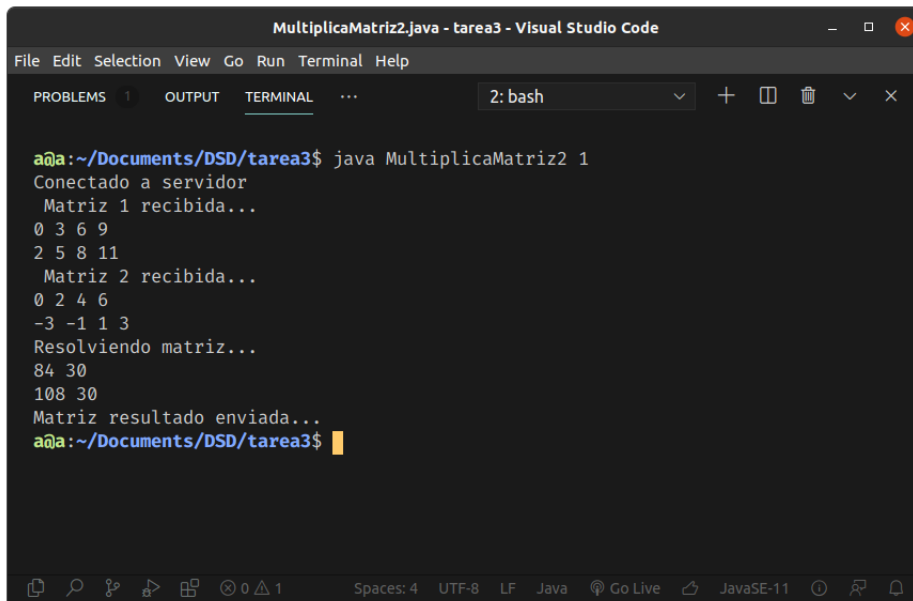
```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 1:java
aaa:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 0
Matriz A
0 3 6 9
2 5 8 11
4 7 10 13
6 9 12 15
Matriz B
0 -3 -6 -9
2 -1 -4 -7
4 1 -2 -5
6 3 0 -3
Matriz B transpuesta
0 2 4 6
-3 -1 1 3
-6 -4 -2 0
-9 -7 -5 -3
Esperando cliente...
```

Al iniciar el programa se muestran las matrices con las matrices con las que se trabajara además aun no se ha conectado ningún cliente al servidor, así que se colocó un texto que nos indica que se están esperando a los clientes y cuando se conecte algún cliente se mostrara un texto que indique que se conectó un cliente.

## 2.2 Ejecución cliente 1

En este punto el servidor ya está iniciado y esperando conexiones, como este es el primer cliente el valor del nodo es 1 y el comando queda como se observa en la imagen.

## Matriz de $N = 4$

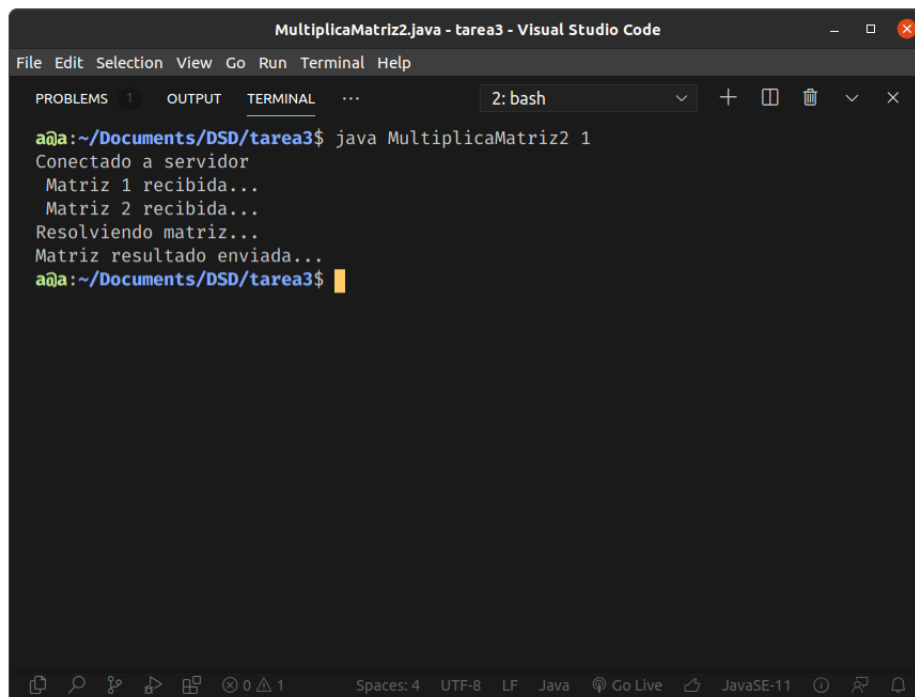


```
MultiplicaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 2: bash
a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 1
Conectado a servidor
Matriz 1 recibida...
0 3 6 9
2 5 8 11
Matriz 2 recibida...
0 2 4 6
-3 -1 1 3
Resolviendo matriz...
84 30
108 30
Matriz resultado enviada...
a@a:~/Documents/DSD/tarea3$
```

En el programa del cliente con  $N = 4$ , se colocó un texto que nos indica cuando el cliente ya está conectado al servidor, mas adelante recibe la primera matriz que envía el servidor, después recibe la segunda matriz, cuando ya tenga las 2 matrices calcula su producto lo muestra y lo manda al servidor.

## Matriz de $N = 1000$

Ahora vemos que el mismo programa cliente pero con  $N = 1000$ , aquí también se colocó un texto que nos indica cuando el cliente ya está conectado al servidor, mas adelante recibe la primera matriz que envía el servidor, después recibe la segunda matriz, cuando ya tenga las 2 matrices calcula su producto lo muestra y lo manda al servidor. La diferencia con el programa de  $N=4$  es que aquí no se despliegan en la consola las matrices por el tamaño de las mismas.

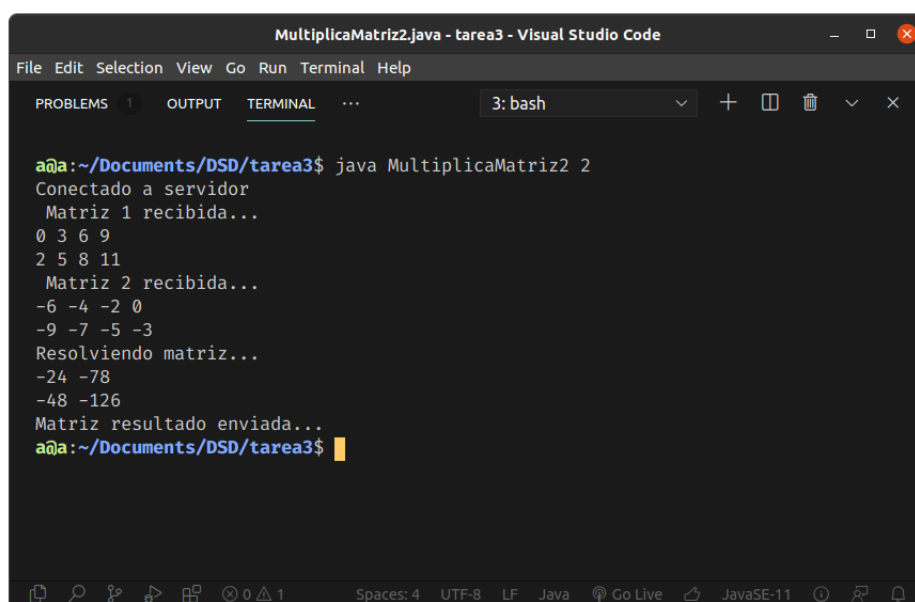


```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 2: bash
a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 1
Conectado a servidor
Matriz 1 recibida...
Matriz 2 recibida...
Resolviendo matriz...
Matriz resultado enviada...
a@a:~/Documents/DSD/tarea3$
```

## 2.3 Ejecución cliente 2

Continuamos con el siguiente nodo de la misma manera en que lo hicimos con el nodo anterior pero como este es el segundo nodo el valor del nodo es 2 y el comando queda como se observa en la imagen.

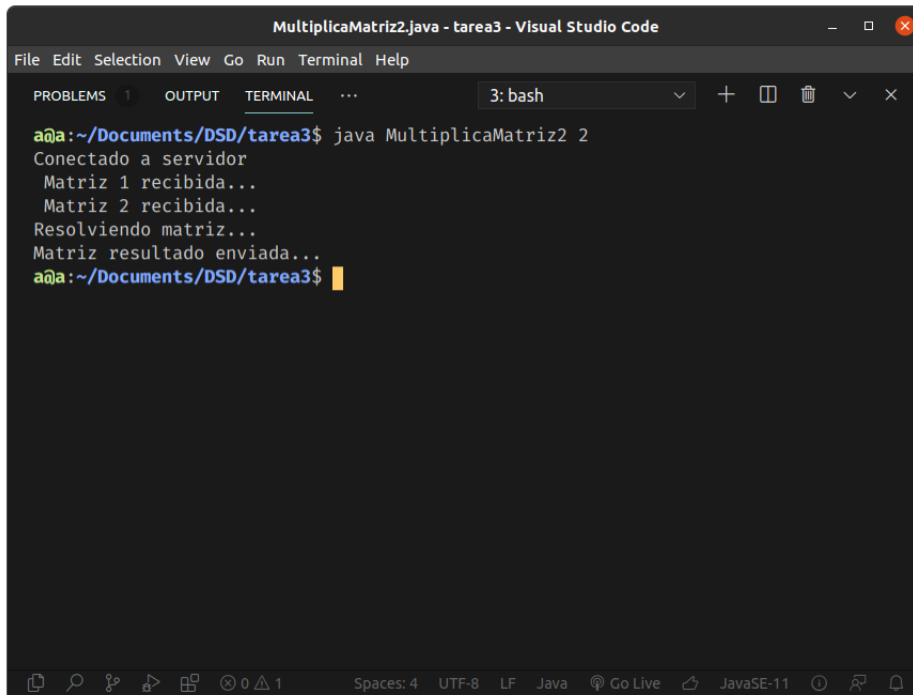
Matriz de  $N = 4$



```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 3: bash
a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 2
Conectado a servidor
Matriz 1 recibida...
0 3 6 9
2 5 8 11
Matriz 2 recibida...
-6 -4 -2 0
-9 -7 -5 -3
Resolviendo matriz...
-24 -78
-48 -126
Matriz resultado enviada...
a@a:~/Documents/DSD/tarea3$
```

Como ya se indico anteriormente en el programa del cliente con  $N=4$  se coloco un texto que nos indica cuando el cliente ya esta conectado al servidor, mas adelante recibe las primer matriz que envía el servidor, después recibe la segunda matriz, cuando ya tenga las 2 matrices calcula su producto lo muestra y lo manda al servidor.

## Matriz de $N = 1000$



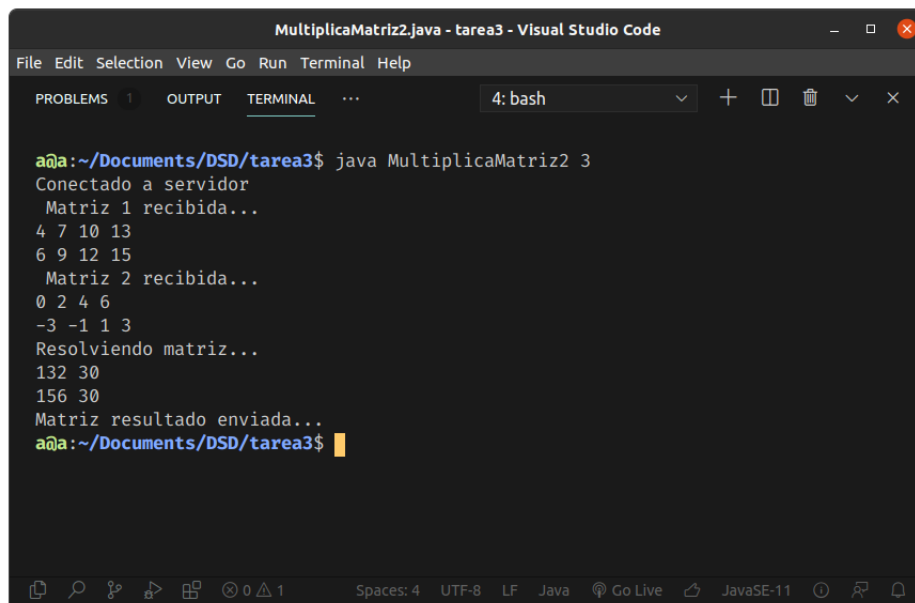
```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 3: bash
a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 2
Conectado a servidor
Matriz 1 recibida...
Matriz 2 recibida...
Resolviendo matriz...
Matriz resultado enviada...
a@a:~/Documents/DSD/tarea3$
```

Ahora vemos que el mismo programa cliente pero con  $N = 1000$ , aquí también se coloco un texto que nos indica cuando el cliente ya esta conectado al servidor, mas adelante recibe las primer matriz que envía el servidor, después recibe la segunda matriz, cuando ya tenga las 2 matrices calcula su producto lo muestra y lo manda al servidor. La diferencia con el programa de  $N=4$  es que aquí no se despliegan en la consola las matrices por el tamaño de las mismas.

## 2.4 Ejecución cliente 3

Ahora continuamos con el nodo tres, de la misma manera en que lo hicimos con el nodo anterior pero como este es el tercero nodo el valor del nodo es 3 y el comando queda como se observa en la imagen.

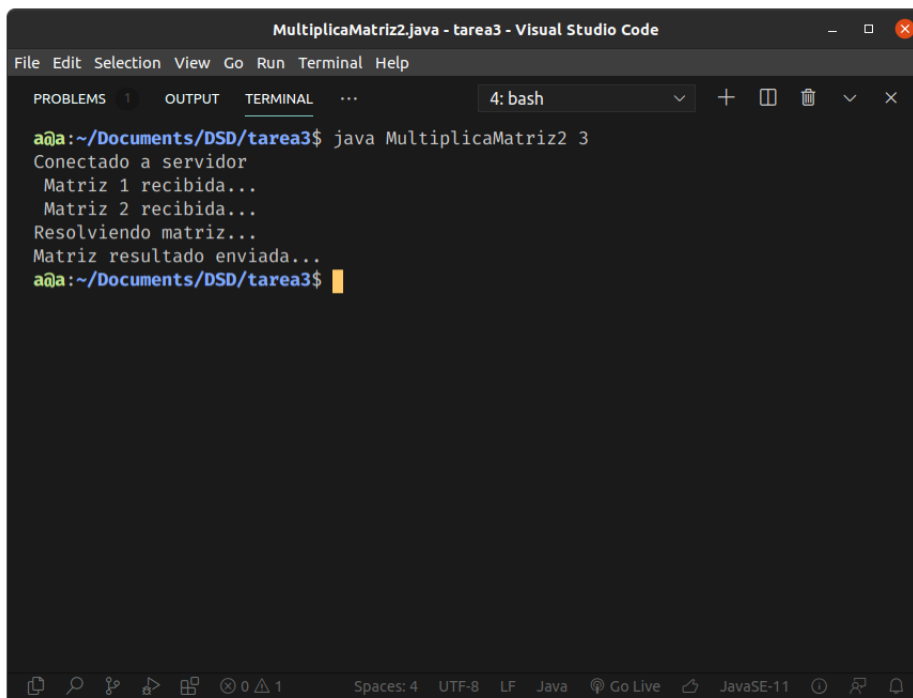
## Matriz de $N = 4$



```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 4: bash
a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 3
Conectado a servidor
Matriz 1 recibida...
4 7 10 13
6 9 12 15
Matriz 2 recibida...
0 2 4 6
-3 -1 1 3
Resolviendo matriz...
132 30
156 30
Matriz resultado enviada...
a@a:~/Documents/DSD/tarea3$
```

De la misma manera a como se hizo en el nodo pasado también se muestra un texto que nos indica cuando del cliente se colocó un texto que nos indica cuando el cliente ya está conectado al servidor, mas adelante recibe la primera matriz que envía el servidor, después recibe la segunda matriz, cuando ya tenga las 2 matrices calcula su producto lo muestra y lo manda al servidor.

## Matriz de $N = 1000$



```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 4: bash
a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 3
Conectado a servidor
Matriz 1 recibida...
Matriz 2 recibida...
Resolviendo matriz...
Matriz resultado enviada...
a@a:~/Documents/DSD/tarea3$
```

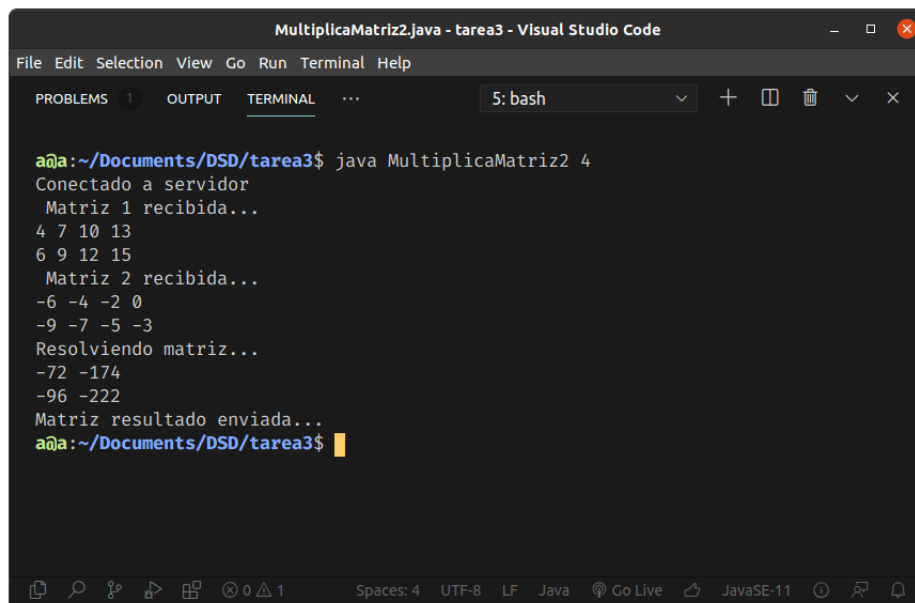
Ahora vemos que el mismo programa cliente pero con  $N = 1000$ , aquí también se coloco un texto que nos indica cuando el cliente ya esta conectado al servidor, mas adelante recibe la primer matriz que envía el servidor, después recibe la segunda matriz, cuando ya tenga las 2 matrices calcula su producto lo muestra y lo manda al servidor. La diferencia con el programa de  $N=4$  es que aquí no se despliegan en la consola las matrices por el tamaño de las mismas.

### 2.5 Ejecución cliente 4

Finalmente tenemos el cuarto nodo y de la misma manera en que lo hicimos con el nodo anterior pero como este es el cuarto nodo el valor del nodo es 4 y el comando queda como se observa en la imagen.



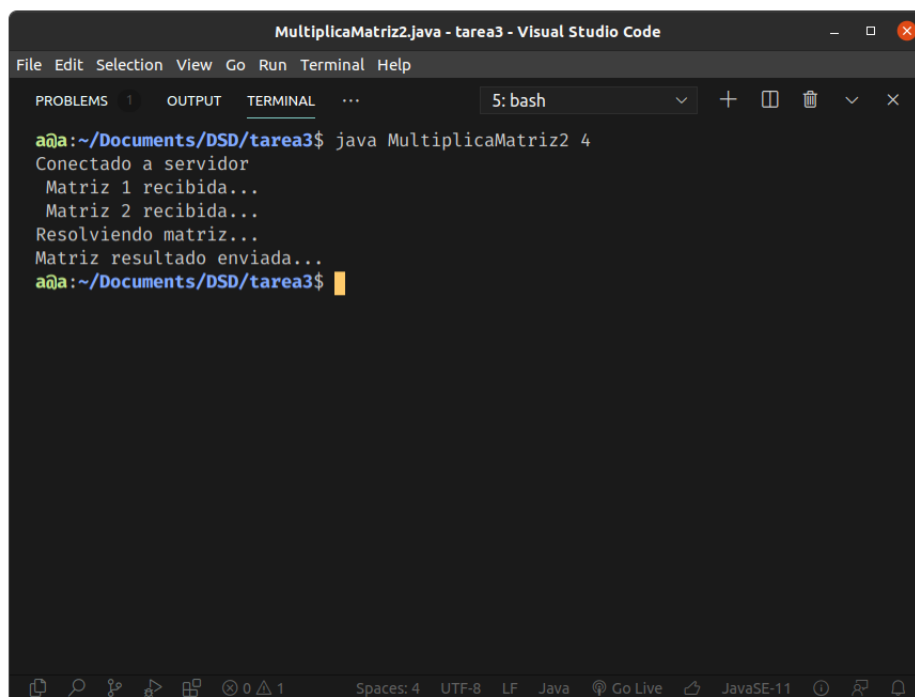
## Matriz de $N = 4$



```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 5: bash
a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 4
Conectado a servidor
Matriz 1 recibida...
4 7 10 13
6 9 12 15
Matriz 2 recibida...
-6 -4 -2 0
-9 -7 -5 -3
Resolviendo matriz...
-72 -174
-96 -222
Matriz resultado enviada...
a@a:~/Documents/DSD/tarea3$
```

Terminamos haciendo por ultima vez el despliegue en la terminal del texto que nos indica cuando el cliente ya esta conectado al servidor, mas adelante recibe las primer matriz que envía el servidor, después recibe la segunda matriz, cuando ya tenga las 2 matrices calcula su producto lo muestra y lo manda al servidor.

## Matriz de $N = 1000$

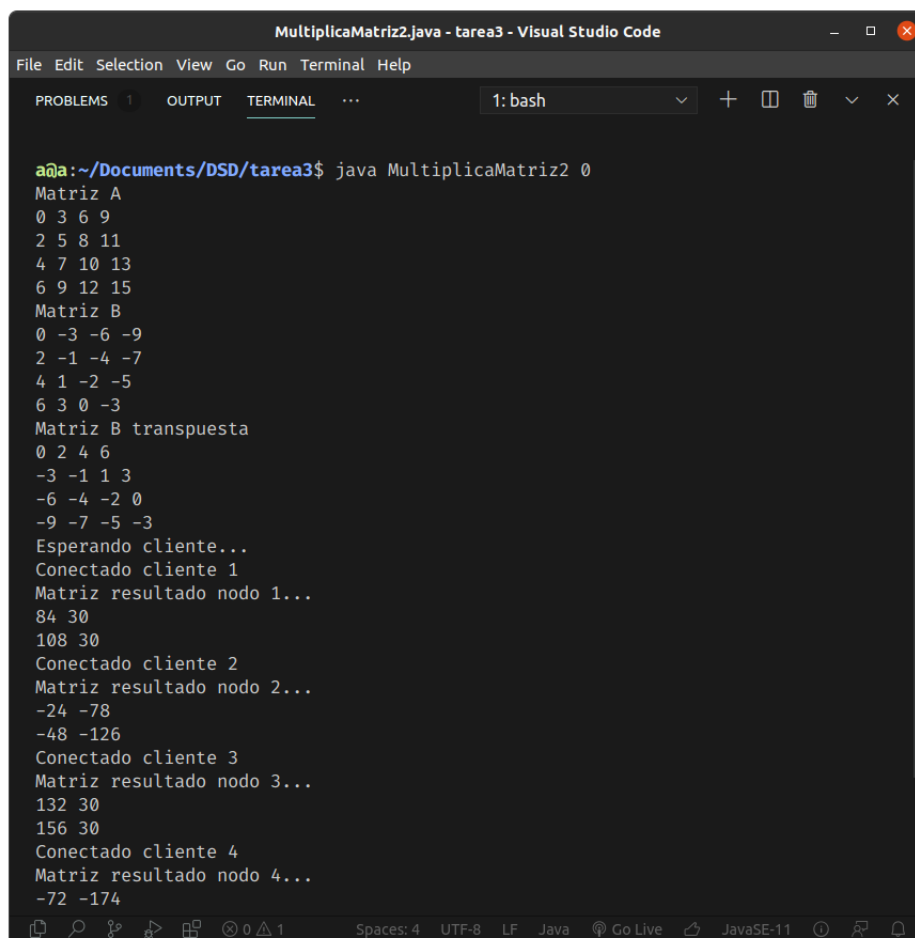


```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 5: bash
a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 4
Conectado a servidor
Matriz 1 recibida...
Matriz 2 recibida...
Resolviendo matriz...
Matriz resultado enviada...
a@a:~/Documents/DSD/tarea3$
```

Ahora vemos que el mismo programa cliente pero con  $N = 1000$ , aquí también se colocó un texto que nos indica cuando el cliente ya está conectado al servidor, mas adelante recibe la primera matriz que envía el servidor, después recibe la segunda matriz, cuando ya tenga las 2 matrices calcula su producto lo muestra y lo manda al servidor. La diferencia con el programa de  $N=4$  es que aquí no se despliegan en la consola las matrices por el tamaño de las mismas.

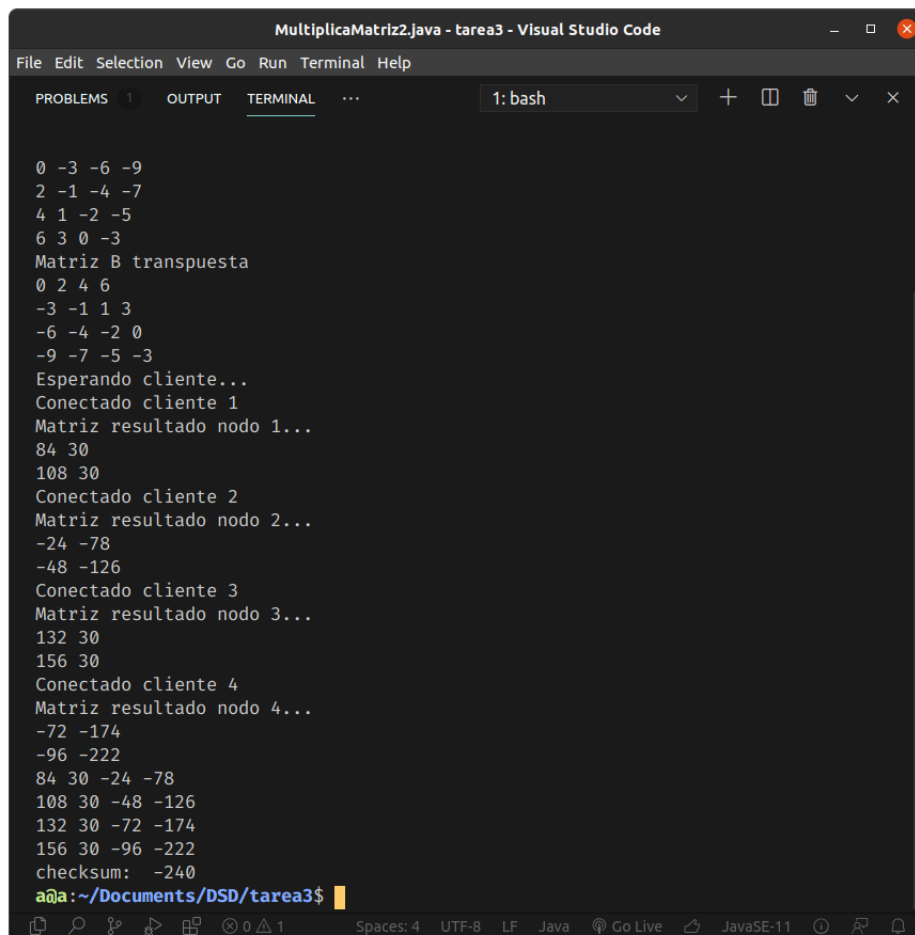
## 2.6 Final de ejecución de servidor y valor de checksum

Matriz de  $N = 4$



```
MultiplicaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 1: bash

a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 0
Matriz A
0 3 6 9
2 5 8 11
4 7 10 13
6 9 12 15
Matriz B
0 -3 -6 -9
2 -1 -4 -7
4 1 -2 -5
6 3 0 -3
Matriz B transpuesta
0 2 4 6
-3 -1 1 3
-6 -4 -2 0
-9 -7 -5 -3
Esperando cliente...
Conectado cliente 1
Matriz resultado nodo 1...
84 30
108 30
Conectado cliente 2
Matriz resultado nodo 2...
-24 -78
-48 -126
Conectado cliente 3
Matriz resultado nodo 3...
132 30
156 30
Conectado cliente 4
Matriz resultado nodo 4...
-72 -174
```

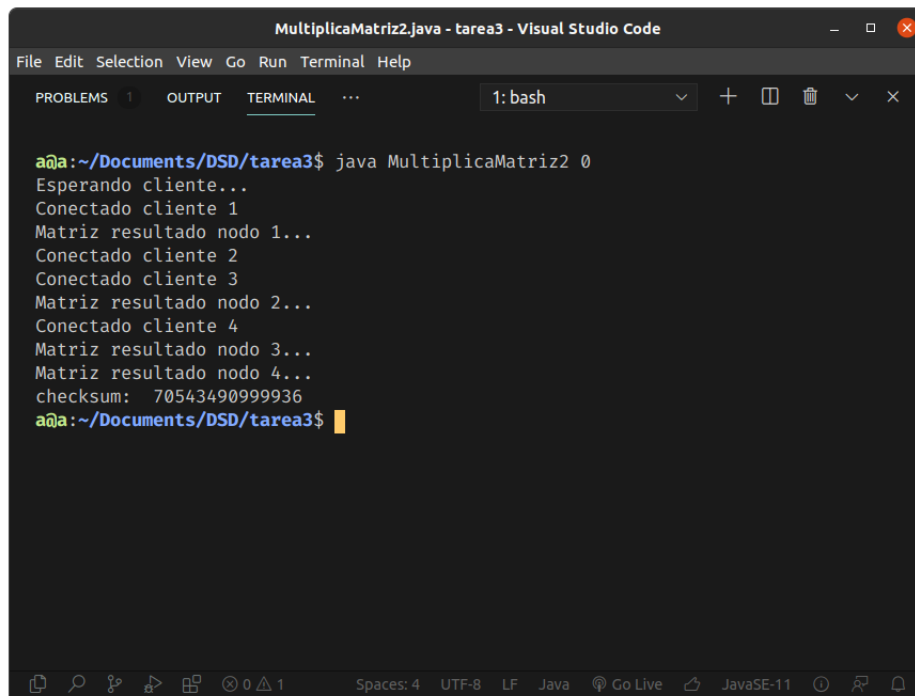


```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 1: bash

0 -3 -6 -9
2 -1 -4 -7
4 1 -2 -5
6 3 0 -3
Matriz B transpuesta
0 2 4 6
-3 -1 1 3
-6 -4 -2 0
-9 -7 -5 -3
Esperando cliente...
Conectado cliente 1
Matriz resultado nodo 1...
84 30
108 30
Conectado cliente 2
Matriz resultado nodo 2...
-24 -78
-48 -126
Conectado cliente 3
Matriz resultado nodo 3...
132 30
156 30
Conectado cliente 4
Matriz resultado nodo 4...
-72 -174
-96 -222
84 30 -24 -78
108 30 -48 -126
132 30 -72 -174
156 30 -96 -222
checksum: -240
a@a:~/Documents/DSD/tarea3$
```

En la terminal del servidor con  $N=4$  podemos observar que se fue indicando el cliente que se iba conectado, la matriz que iba mandando a cada cliente y al final cuando todos los cliente terminaron de hacer su calculo entregaron su resultado al servidor el cual lo sumo y entrego el valor final de del checksum y de la multiplicación de las matrices, el cual se muestra al final de la ejecución del servidor.

## Matriz de $N = 1000$



```
MultiplcaMatriz2.java - tarea3 - Visual Studio Code
File Edit Selection View Go Run Terminal Help
PROBLEMS 1 OUTPUT TERMINAL ... 1: bash
a@a:~/Documents/DSD/tarea3$ java MultiplicaMatriz2 0
Esperando cliente...
Conectado cliente 1
Matriz resultado nodo 1...
Conectado cliente 2
Conectado cliente 3
Matriz resultado nodo 2...
Conectado cliente 4
Matriz resultado nodo 3...
Matriz resultado nodo 4...
checksum: 70543490999936
a@a:~/Documents/DSD/tarea3$
```

En la terminal del servidor con  $N=1000$  podemos observar que se fue indicando el cliente que se iba conectado, las operaciones que se van realizando, cuando todos los cliente terminaron de hacer su calculo entregaron su resultado al servidor el cual lo sumo y entrego el valor final de del checksum y de la multiplicación de las matrices, el cual se muestra al final de la ejecución del servidor.

## 3 Conclusiones

Con lo desarrollado en la tarea pude observar que desarrollar una solución distribuida es mas tardado y laborioso que crear un programa secuencial ya que para solucionar la multiplicación de 2 matrices en un programa no distribuido resulta mas sencillo desarrollar el código pero para un programa distribuido existen algunas aspectos que complican el desarrolló del código aun siendo el mismo algoritmo, las complicaciones se deben a la coordinación que debe existir entre el servidor y los nodos clientes, pero al final del desarrollo del código distribuido se logra observar que ahora se obtiene una solución mas rápida esto se nota mas con la matriz de  $N=1000$ , eso se debe a que se tiene mas de un proceso trabajando en la solución.