

Informe Laboratorio de Evaluación de Rendimiento

Sistemas Operativos



Integrantes:

Maria Paula Rodríguez Ruiz

Profesor:

John Corredor

Dpto. de Ingeniería de Sistemas

Pontificia Universidad Javeriana

Bogotá D.C.

31 de Octubre del 2024

El presente informe hablara respecto a: El comando SSH para la conexión remota con dos máquinas virtuales establecidas y una comparativa entre los dos sistemas de cómputo mediante la ejecución de diferentes comandos de los cuales se reunieron sus resultados en una hoja de calculo para su posterior análisis final.

1. Conexión SSH

Para conectarme de forma remota a las máquinas virtuales con IP '10.43.103.133' y IP '10.43.101.140', utilicé la VPN de arpuj.javeriana que permite acceder a la red privada de la institución. Posterior a ello, la conexión remota se realizó de forma segura utilizando el protocolo SSH, el cual permite ejecutar comandos y administrar servidores a distancia. Los comandos utilizados fueron "ssh estudiante@10.43.103.133" y "ssh estudiante@10.43.101.140", comandos que se ejecutaron en el puerto predeterminado 22 y que permiten establecer la conexión bajo el usuario estudiante. Posterior a él se ingresaron las contraseñas correspondientes y se establecieron las conexiones como se muestra a continuación en la figura 1 y 2.

```
PS C:\Users\thega> ssh estudiante@10.43.103.133
The authenticity of host '10.43.103.133 (10.43.103.133)' can't be established.
ED25519 key fingerprint is SHA256:Rrw482RfHCGbWYjisfJl7jxCfV7aLXdPMaTFYv8hjrK.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.43.103.133' (ED25519) to the list of known hosts.
estudiante@10.43.103.133's password:
Last login: Wed Sep 25 11:44:38 2024 from 10.195.48.108
```

Figura 1. Comando protocolo SSH maquina 10.43.103.133. Elaboración propia

```
PS C:\Users\thega> ssh estudiante@10.43.101.140
The authenticity of host '10.43.101.140 (10.43.101.140)' can't be established.
ED25519 key fingerprint is SHA256:/trx9KmNHmQLgvEfn702hhak29kAUxHEnKjP4UhyKpQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.43.101.140' (ED25519) to the list of known hosts.
estudiante@10.43.101.140's password:
Permission denied, please try again.
estudiante@10.43.101.140's password:
Permission denied, please try again.
estudiante@10.43.101.140's password:
estudiante@10.43.101.140: Permission denied (publickey,password).
PS C:\Users\thega> ssh estudiante@10.43.101.140
estudiante@10.43.101.140's password:
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-40-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

Expanded Security Maintenance for Applications is not enabled.

165 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

6 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

*** System restart required ***
Last login: Mon Aug  5 16:54:15 2024 from 10.4.100.111
```

Figura 2. Comando protocolo SSH maquina 10.43.101.140. Elaboración propia

2. Comandos para el monitoreo del sistema

a. Comando cat /etc/os-release

El comando cat /etc/os-release proporciona una visualización de cual es el sistema operativo que están usando nuestras máquinas virtuales como se muestra a continuación en la figura 3 y 4.

```
[estudiante@ING-PDGE09 Evaluacion]$ cat /etc/os-release
NAME="Rocky Linux"
VERSION="9.4 (Blue Onyx)"
ID="rocky"
ID_LIKE="rhel centos fedora"
VERSION_ID="9.4"
PLATFORM_ID="platform:el9"
PRETTY_NAME="Rocky Linux 9.4 (Blue Onyx)"
ANSI_COLOR="0;32"
LOGO="fedora-logo-icon"
CPE_NAME="cpe:/o:rocky:rocky:9::baseos"
HOME_URL="https://rockylinux.org/"
BUG_REPORT_URL="https://bugs.rockylinux.org/"
SUPPORT_END="2032-05-31"
ROCKY_SUPPORT_PRODUCT="Rocky-Linux-9"
ROCKY_SUPPORT_PRODUCT_VERSION="9.4"
REDHAT_SUPPORT_PRODUCT="Rocky Linux"
REDHAT_SUPPORT_PRODUCT_VERSION="9.4"
```

Figura 3. Comando cat /etc/os-release maquina 10.43.103.133. Elaboración propia

```
estudiante@ing-gen304:~/Documents/Evaluacion$ cat /etc/os-release
PRETTY_NAME="Ubuntu 24.04 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
VERSION="24.04 LTS (Noble Numbat)"
VERSION_CODENAME=noble
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=noble
LOGO=ubuntu-logo
```

Figura 4. Comando cat /etc/os-release maquina 10.43.101.140. Elaboración propia

Este comando es útil para conocer realmente que estamos comparando, en este caso sería para la maquina '10.43.103.133' un Rocky Linux fedora y para la maquina '10.43.101.140' un Ubuntu debian.

b. Comando top y 1

El comando `top` seguido de la tecla `1` proporcionan una visualización en tiempo real de los procesos activos y el uso de recursos del sistema, en especial el uso de la CPU, y al presionar `1`, se desglosa el uso de cada núcleo de la CPU individualmente como se muestra a continuación en la figura 5 y 6.

```
top - 10:44:41 up 98 days, 20:53, 1 user, load average: 0.00, 0.02, 0.00
Tasks: 259 total, 1 running, 258 sleeping, 0 stopped, 0 zombie
%Cpu0  :  0.3 us,  0.0 sy,  0.0 ni, 99.3 id,  0.0 wa,  0.3 hi,  0.0 si,  0.0 st
%Cpu1  :  0.0 us,  0.0 sy,  0.0 ni, 99.7 id,  0.0 wa,  0.3 hi,  0.0 si,  0.0 st
%Cpu2  :  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu3  :  0.3 us,  0.0 sy,  0.0 ni, 99.3 id,  0.0 wa,  0.3 hi,  0.0 si,  0.0 st
MiB Mem : 11673.0 total, 3985.0 free, 3310.7 used, 4780.7 buff/cache
MiB Swap: 4100.0 total, 4100.0 free,  0.0 used. 8362.3 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
3886847	root	20	0	2623844	319700	102244	S	1.0	2.7	441:57.34	ampdaemon
1	root	20	0	175032	18744	10736	S	0.0	0.2	1:43.99	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:03.52	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	slub_flushwq
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns

Figura 5. Comando top y 1 10.43.103.133. Elaboración propia

```
top - 10:04:33 up 69 days, 18:37, 3 users, load average: 0.04, 0.08, 0.05
Tasks: 382 total, 1 running, 381 sleeping, 0 stopped, 0 zombie
%Cpu0  :  1.7 us,  1.3 sy,  0.0 ni, 97.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu1  :  0.3 us,  0.7 sy,  0.0 ni, 99.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu2  :  0.7 us,  1.0 sy,  0.0 ni, 98.3 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
%Cpu3  :  0.7 us,  0.3 sy,  0.0 ni, 99.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem : 11914.2 total, 2359.7 free, 2755.0 used, 7311.5 buff/cache
MiB Swap: 4096.0 total, 4096.0 free,  0.0 used. 9159.1 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
600690	estudia+	20	0	117196	9984	7296	S	3.0	0.1	1:08.95	pipewire
601326	estudia+	20	0	1628760	295500	192124	S	2.3	2.4	0:57.73	gnome-remote-de
1198	root	20	0	17552	7808	6912	S	0.3	0.1	0:24.63	systemd-machine
1241	root	20	0	558004	19812	16496	S	0.3	0.2	30:48.10	NetworkManager
1476	zabbix	20	0	31980	5056	3456	S	0.3	0.0	21:51.59	zabbix_agentd
1693	gdm	20	0	4780112	232828	126444	S	0.3	1.9	245:53.93	gnome-shell
600918	estudia+	20	0	5181000	754108	292252	S	0.3	6.2	0:58.28	gnome-shell
601202	estudia+	20	0	140848	37504	29824	S	0.3	0.3	0:09.73	vmtoolsd
602483	estudia+	20	0	14556	5888	3712	R	0.3	0.0	0:00.13	top
1	root	20	0	23464	14848	9600	S	0.0	0.1	5:09.65	systemd

Figura 6. Comando top y 1 10.43.101.140. Elaboración propia

Este comando es extremadamente útil para el monitoreo y diagnóstico del rendimiento del sistema, ya que permite identificar procesos que

consumen muchos recursos y ajustar la carga para mejorar el rendimiento. Además nos permite acceder a información como que: la maquina '10.43.103.133' tiene 4 Cpu's y 11673 MiB de memoria RAM mientras que la maquina '10.43.101.140' tiene 4 Cpu's y 11914 MiB de memoria RAM

c. Comando `lscpu`

El comando ``lscpu`` detalla la arquitectura y los componentes de la CPU, incluyendo la cantidad de núcleos, hilos y tamaños de caché, entre otros como se puede ver a continuación en la figura 7 y 8.

```
[estudiante@ING-PDGE09 ~]$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Address sizes:          43 bits physical, 48 bits virtual
Byte Order:             Little Endian
CPU(s):                 4
On-line CPU(s) list:    0-3
Vendor ID:              GenuineIntel
Model name:             Intel(R) Xeon(R) Gold 6240R CPU @ 2.40GHz
CPU family:             6
Model:                  85
Thread(s) per core:     1
Core(s) per socket:     1
Socket(s):              4
Stepping:               7
BogoMIPS:               4788.74
Flags:                  fpu vme de pse tsc msr pae mce cx8 apic sep mt
                        n nopl xtopology tsc_reliable nonstop_tsc cpui
                        xsave avx f16c rdrand hypervisor lahf_lm abm 3
                        12dq rdseed adx smap clflushopt clwb avx512cd
Virtualization features:
Hypervisor vendor:      VMware
Virtualization type:    full
Caches (sum of all):
L1d:                    128 KiB (4 instances)
L1i:                    128 KiB (4 instances)
L2:                     4 MiB (4 instances)
L3:                     143 MiB (4 instances)
NUMA:
NUMA node(s):           1
NUMA node0 CPU(s):      0-3
Vulnerabilities:
Gather data sampling:    Unknown: Dependent on hypervisor status
Itlb multihit:           KVM: Mitigation: VMX unsupported
L1tf:                   Not affected
Mds:                     Not affected
Meltdown:               Not affected
Mmio stale data:         Vulnerable: Clear CPU buffers attempted, no mi
Retbleed:                Mitigation; Enhanced IBRS
Spec rstack overflow:    Not affected
Spec store bypass:       Mitigation; Speculative Store Bypass disabled
Spectre v1:              Mitigation; usercopy/swapgs barriers and __use
Spectre v2:              Mitigation; Enhanced / Automatic IBRS, IBPB co
Srbds:                   Not affected
Tsx async abort:         Not affected
[estudiante@ING-PDGE09 ~]$
```

Figura 7. Comando `lscpu` 10.43.103.133. Elaboración propia

```

estudiante@ing-gen304:~/Documents/Evaluacion$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Address sizes:          43 bits physical, 48 bits virtual
Byte Order:             Little Endian
CPU(s):                 4
On-line CPU(s) list:    0-3
Vendor ID:              GenuineIntel
Model name:             Intel(R) Xeon(R) Gold 6348 CPU @ 2.60GHz
CPU family:             6
Model:                 85
Thread(s) per core:     1
Core(s) per socket:     1
Socket(s):              4
Stepping:               7
BogoMIPS:               5187.81
Flags:                  fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov
                        pat pse36 clflush mmx fxsr sse sse2 ss syscall nx pdpe1gb rd
                        tscp lm constant_tsc arch_perfmon nopl xtopology tsc_reliable
                        nonstop_tsc cpuid tsc_known_freq pni pclmulqdq ssse3 fma cx1
                        6 pcid sse4_1 sse4_2 x2apic movbe popcnt tsc_deadline_timer a
                        es xsave avx f16c rdrand hypervisor lahf_lm abm 3dnowprefetch
                        ssbd ibrs ibpb stibp ibrs_enhanced fsgsbase tsc_adjust bmi1
                        avx2 smep bmi2 invpcid avx512f avx512dq rdseed adx smap clflu
                        shopt clwb avx512cd avx512bw avx512vl xsaveopt xsavec xsave
                        arat pku ospke md_clear flush_lld arch_capabilities

Virtualization features:
Hypervisor vendor:      VMware
Virtualization type:    full
Caches (sum of all):
L1d:                    192 KiB (4 instances)
L1i:                    128 KiB (4 instances)
L2:                     5 MiB (4 instances)
L3:                     168 MiB (4 instances)
NUMA:
NUMA node(s):           1
NUMA node0 CPU(s):      0-3
Vulnerabilities:
Gather data sampling:    Unknown: Dependent on hypervisor status
Itlb multihit:           KVM: Mitigation: VMX unsupported
L1tf:                    Not affected
Mds:                     Not affected
Meltdown:                Not affected
Mmio stale data:         Vulnerable: Clear CPU buffers attempted, no microcode; SMT Ho
                        st state unknown
Reg file data sampling:  Not affected
Retbleed:                Mitigation; Enhanced IBRS
Spec rstack overflow:    Not affected
Spec store bypass:       Mitigation; Speculative Store Bypass disabled via prctl
Spectre v1:              Mitigation; usercopy/swapgs barriers and __user pointer sanit
                        ization
Spectre v2:              Mitigation; Enhanced / Automatic IBRS; IBPB conditional; RSB
                        filling; PBRSE-eIBRS SW sequence; BHI SW loop, KVM SW loop
Srbds:                   Not affected
Tsx async abort:         Not affected

```

Figura 8. Comando `lscpu` 10.43.101.140. Elaboración propia

Este comando proporciona una vista general de la configuración del hardware del procesador, lo cual es esencial para análisis de rendimiento y configuración de software optimizada para la CPU disponible. Además nos permite acceder a información como que ambas máquinas virtuales tienen 4 instancias de memoria L1d, L1i, L2 y L3.

d. Jerarquía de memoria

Con la información accedida hasta ahora es posible construir la jerarquía de memoria de la máquina virtual y se ve como la figura 9 y 10.

(VM) 10.43.103.133				
L1d	128kiB	128kiB	128kiB	128kiB
CPUs	0	1	2	3
L1i	128kiB	128kiB	128kiB	128kiB
L2	4MiB	4MiB	4MiB	4MiB
L3	143MiB	143MiB	143MiB	143MiB
RAM	11673 MiB			

Figura 9. Jerarquía de memoria 10.43.103.133. Elaboración propia

(VM) 10.43.101.140				
L1d	192kiB	192kiB	192kiB	192kiB
CPUs	0	1	2	3
L1i	128kiB	128kiB	128kiB	128kiB
L2	5MiB	5MiB	5MiB	5MiB
L3	168MiB	168MiB	168MiB	168MiB
RAM	11914 MiB			

Figura 10. Jerarquía de memoria 10.43.101.140. Elaboración propia

De ella, es preciso denotar que ambos sistemas de cómputo de las máquinas virtuales tienen un buen rendimiento pues que cada procesador tiene su propia instancia L1d, L1i, L2 y L3 por lo que el balanceo de cargas está super bien elaborado.

e. Comando df -h

El comando `df -h` muestra el uso del espacio en disco de manera legible para humanos (MB, GB). Esto es especialmente útil para monitorear el almacenamiento disponible y el utilizado como se puede ver a continuación en la figura 11 y 12.

```
[estudiante@ING-PDGE09 ~]$ df -h
Filesystem                Size      Used Avail Use% Mounted on
devtmpfs                   4.0M         0   4.0M   0% /dev
tmpfs                       5.7G         0   5.7G   0% /dev/shm
tmpfs                       2.3G      38M   2.3G   2% /run
/dev/mapper/rl_plantillarocky9-root 24G     3.3G    21G  14% /
/dev/sda2                   960M     389M   572M  41% /boot
/dev/sda1                  1022M     7.1M  1015M   1% /boot/efi
/dev/mapper/rl_plantillarocky9-var  15G     1.4G    14G   9% /var
/dev/mapper/rl_plantillarocky9-home  15G    140M    15G   1% /home
10.43.103.140:/almacen      24G     5.8G    19G  25% /almacen
tmpfs                       1.2G     4.0K    1.2G   1% /run/user/1001
[estudiante@ING-PDGE09 ~]$ |
```

Figura 11. Comando df -h 10.43.103.133. Elaboración propia

```
estudiante@ing-gen304:~/Documents/Evaluacion$ df -h
Filesystem                Size      Used Avail Use% Mounted on
tmpfs                      1.2G      2.3M   1.2G   1% /run
/dev/sda2                   79G      16G    59G  22% /
tmpfs                       5.9G         0   5.9G   0% /dev/shm
tmpfs                       5.0M         0   5.0M   0% /run/lock
tmpfs                       5.9G    220K   5.9G   1% /run/qemu
tmpfs                       1.2G    104K    1.2G   1% /run/user/120
tmpfs                       1.2G    124K    1.2G   1% /run/user/1000
```

Figura 12. Comando df -h 10.43.101.140. Elaboración propia

Este comando permite identificar problemas de almacenamiento y posibles cuellos de botella relacionados con la falta de espacio en disco, lo que podría afectar el rendimiento de la máquina. Además nos permite acceder a información como que: la máquina virtual 1 tiene un

almacenamiento total de 24G en el disco duro o SSD (gracias a la raíz “/”) y la máquina virtual 2 tiene un almacenamiento total de 79G en el disco duro o SSD (gracias a la raíz “/”); ficheros que no se guardaran al apagar el hardware.

3. Evaluación de rendimiento

Ahora bien, para la evaluación de rendimiento se compilaron los programas mm_clasico.c y mm_transpuesta.c que representan algoritmos de la multiplicación de matrices, el primero es el clásico filas por columnas de la matriz A por la matriz B, pero el segundo es filas x filas pues es la matriz A por la transpuesta de la matriz B pero implementando posix para distribuir las cargas de trabajo de los procesos y devolviendo como resultado el tiempo de ejecución en microsegundos (us), por lo que el primer paso fue:

1. Compilar los programas:

```
gcc mm_clasica.c -o clasica
```

```
gcc mm_transpuesta.c -o transpuesta
```

Y se ejecutaron de la siguiente forma: Cada programa se ejecutó para una matriz de 800x800, 1000x1000 y 1800x18000, pero 4 veces por 1 hilo, 4 veces por 2 hilos y 4 veces por 4 hilos como se puede ver a continuación en las figuras 13 a 18 para la maquina 1 con IP: 10.43.103.133 y en las figuras 19 a 24 para la maquina 2 con IP 10.43.101.140:

```

[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 1      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 1
:-> 2186194 µs                                           :-> 1648226 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 1      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 1
:-> 1670141 µs                                           :-> 1924254 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 1      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 1
:-> 2158564 µs                                           :-> 2278915 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 1      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 1
:-> 1824474 µs                                           :-> 1488839 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 2      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 2
:-> 1752541 µs                                           :-> 1244033 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 2      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 2
:-> 1590908 µs                                           :-> 909888 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 2      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 2
:-> 1290491 µs                                           :-> 739964 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 2      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 2
:-> 1126532 µs                                           :-> 1440982 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 4      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 4
:-> 1085611 µs                                           :-> 1150177 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 4      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 4
:-> 1112522 µs                                           :-> 674991 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 4      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 4
:-> 547297 µs                                           :-> 983961 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 800 4      [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 800 4
:-> 880375 µs                                           :-> 468680 µs

```

Figura 13 y 14. Ejecuciones en la maquina 1: 10.43.103.133. Elaboración propia

```

[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 1    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 1
:-> 3861889 µs                                           :-> 3086376 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 1    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 1
:-> 3510798 µs                                           :-> 2745074 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 1    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 1
:-> 3891596 µs                                           :-> 3289093 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 1    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 1
:-> 3180961 µs                                           :-> 3163290 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 2    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 2
:-> 1564371 µs                                           :-> 1944525 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 2    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 2
:-> 2377613 µs                                           :-> 1962109 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 2    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 2
:-> 1952692 µs                                           :-> 1389578 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 2    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 2
:-> 1829519 µs                                           :-> 1827174 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 4    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 4
:-> 1470293 µs                                           :-> 864098 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 4    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 4
:-> 1484755 µs                                           :-> 1000183 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 4    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 4
:-> 1565398 µs                                           :-> 776233 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1000 4    [estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1000 4
:-> 1579088 µs                                           :-> 1624538 µs

```

Figura 15 y 16. Ejecuciones en la maquina 1: 10.43.103.133. Elaboración propia

```

[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 1
:-> 24041845 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 1
:-> 23390697 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 1
:-> 23659388 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 1
:-> 23225909 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 2
:-> 11889601 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 2
:-> 11975587 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 2
:-> 12073309 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 2
:-> 12059906 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 4
:-> 6204882 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 4
:-> 5851865 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 4
:-> 5827741 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./clasico 1800 4
:-> 6327882 µs

[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 1
:-> 16173335 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 1
:-> 16154760 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 1
:-> 15639796 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 1
:-> 16332517 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 2
:-> 7893842 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 2
:-> 8394787 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 2
:-> 8573250 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 2
:-> 8553866 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 2
:-> 8362119 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 4
:-> 3983751 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 4
:-> 4537373 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 4
:-> 4392507 µs
[estudiante@ING-PDGE09 Evaluacion]$ ./transpuesta 1800 4
:-> 4806422 µs

```

Figura 17 y 18. Ejecuciones en la maquina 1: 10.43.103.133. Elaboración propia

```

estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 1
:-> 3156821 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 1
:-> 2811809 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 1
:-> 2816356 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 1
:-> 2135352 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 2
:-> 1853097 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 2
:-> 1831223 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 2
:-> 1836878 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 2
:-> 1854939 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 4
:-> 678381 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 4
:-> 1022232 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 4
:-> 1295281 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 800 4
:-> 1557421 µs

estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 1
:-> 2451464 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 1
:-> 2659474 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 1
:-> 2314168 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 1
:-> 2289335 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 2
:-> 1449880 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 2
:-> 1188320 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 2
:-> 934889 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 2
:-> 1686528 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 4
:-> 1277738 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 4
:-> 658080 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 4
:-> 950404 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 800 4
:-> 1127457 µs

```

Figura 19 y 20. Ejecuciones en la maquina 2: 10.43.101.140. Elaboración propia

```

estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 1
:-> 3993465 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 1
:-> 5036997 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 1
:-> 3871330 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 1
:-> 4500973 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 2
:-> 2682342 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 2
:-> 2448888 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 2
:-> 2653295 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 2
:-> 2879390 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 4
:-> 1722394 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 4
:-> 1476469 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 4
:-> 1106676 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1000 4
:-> 1876850 µs

estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 1
:-> 3877106 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 1
:-> 3901296 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 1
:-> 2874506 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 1
:-> 3977227 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 2
:-> 1949610 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 2
:-> 2042873 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 2
:-> 2618308 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 2
:-> 2172291 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 4
:-> 970216 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 4
:-> 1559822 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 4
:-> 1132878 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1000 4
:-> 1771613 µs

```

Figura 21 y 22. Ejecuciones en la maquina 2: 10.43.101.140. Elaboración propia

```

estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 1
:-> 22478225 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 1
:-> 23194842 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 1
:-> 22417413 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 1
:-> 26180127 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 2
:-> 13869849 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 2
:-> 12785573 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 2
:-> 13392424 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 2
:-> 11784040 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 4
:-> 6150820 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 4
:-> 6309033 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 4
:-> 6641342 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./clasico 1800 4
:-> 5980220 µs

estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 1
:-> 17813813 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 1
:-> 17049349 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 1
:-> 16758705 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 1
:-> 18275506 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 2
:-> 8771735 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 2
:-> 10200804 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 2
:-> 8777976 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 2
:-> 9457812 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 4
:-> 5613088 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 4
:-> 5346886 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 4
:-> 4887927 µs
estudiante@ing-gen304:~/Documents/Evaluacion$ ./transpuesta 1800 4
:-> 5529021 µs

```

Figura 23 y 24. Ejecuciones en la maquina 2: 10.43.101.140. Elaboración propia

4. Conclusión

Posterior a la ejecución de los programas según la planeación planteada se reunieron sus resultados en una tabla mediante una hoja de cálculo resultando en la figura 25:

presentada a continuación:

	MAQUINA 1: 10.43.103.133						MAQUINA 2: 10.43.101.140					
	MM_CLASICA			MM_TRANSPUESTA			MM_CLASICA			MM_TRANSPUESTA		
	1 HILO	2 HILOS	4 HILOS	1 HILO	2 HILOS	4 HILOS	1 HILO	2 HILOS	4 HILOS	1 HILO	2 HILOS	4 HILOS
800	2186194	1752541	1085611	1648226	1244033	1150177	3156821	1853097	678381	2451464	1449880	1277738
	1670141	1590908	1112522	1924254	909888	674991	2811809	1831223	1022232	2659474	1188320	658080
	2158564	1290491	547297	2278915	739964	983961	2816356	1836878	1295281	2314168	934889	950404
	1824474	1126532	880375	1488839	1440982	468680	2135352	1854939	1557421	2289335	1686528	1127457
AVG	1959843.25	1440118	906451.25	1835058.5	1083716.75	819452.25	2730084.5	1844034.25	1138328.75	2428610.25	1314904.25	1003419.75
1000	3861889	1564371	1470293	3086376	1944525	864098	3993465	2682342	1722394	3877106	1949610	970216
	3510798	2377613	1484755	2745074	1962109	1000183	5036997	2448888	1476469	3901296	2042873	1559822
	3891596	1952692	1565398	3289093	1389578	776233	3871330	2653295	1106676	2874506	2618308	1132878
	3180961	1829519	1579088	3163290	1827174	1624538	4500973	2879390	1876850	3977227	2172291	1771613
AVG	3611311	1931048.75	1524883.5	3070958.25	1780846.5	1066263	4350691.25	2665978.75	1545597.25	3657533.75	2195770.5	1358632.25
1800	24041845	11889601	6204882	16173335	7893842	3983751	22478225	13869849	6150820	17813813	8771735	5613088
	23390697	11975587	5851865	16154760	8394787	4537373	23194842	12785573	6309033	17049349	10200804	5346886
	23659388	12073309	5827741	15639796	8573250	4392507	22417413	13392424	6641342	16758705	8777976	4887927
	23225909	12059906	6327882	16332517	8553866	4806422	26180127	11784040	5980220	18275506	9457812	5529021
AVG	23579459.75	11999600.75	6053092.5	16075102	8353936.25	4430013.25	23567651.75	12957971.5	6270353.75	17474343.25	9302081.75	5344230.5

Figura 25. Tabla reunión de datos. Elaboración propia

Donde cómo se puede precisar, se aplicó estadística a la tabla para calcular el promedio de las 4 iteraciones por cada programa pues nunca un evento es igual a otro más sin embargo el promedio me permite acercarme a un valor que determine la realidad. Este se calculó en base a: el tamaño de matriz, el número de hilos y la maquina correspondiente.

Y a partir de ella, podemos decir que:

1. En la **Máquina 1**, los tiempos de ejecución con la multiplicación clásica tienden a ser menores a medida que se aumentan los hilos, aunque con menos eficiencia cuando se pasa de 2 a 4 hilos, especialmente en matrices más grandes como la de 1800x1800.
2. En la **Máquina 1** con multiplicación transpuesta, también hay una disminución de tiempo al incrementar los hilos, pero esta reducción parece ser menos significativa en matrices pequeñas (800x800), mostrando un mejor rendimiento con matrices más grandes.
3. **Máquina 2** presenta patrones similares en términos de reducción de tiempo con el aumento de hilos, pero los tiempos absolutos son generalmente más bajos en

comparación con Máquina 1.

Posteriormente, creamos dos tablas que nos permitieran sacar conclusiones más precisas presentadas a continuación en la figura 26 y 27:

1. Tabla Comparativa de Tiempos entre Máquinas por Tipo de Multiplicación: Esta tabla muestra los tiempos promedio en ambas máquinas en función del tamaño de la matriz y tipo de multiplicación.

Tamaño de Matriz	Tipo de Multiplicación	Máquina 1 (Promedio, us)	Máquina 2 (Promedio, us)
800x800	Clásica	1435470.833	1904149.17
1000x1000	Clásica	2355747.75	2854089.08
1800x1800	Clásica	13877384.33	14265325.7
800x800	Transpuesta	1246075.833	1582311.42
1000x1000	Transpuesta	1972689.25	2403978.83
1800x1800	Transpuesta	9619683.833	10706885.2

Figura 26. Tabla comparativa de tiempos entre maquinas por tipo de multiplicación. Elaboración propia

2. Tabla de Tiempos Promedio por Máquina, Tipo de Multiplicación y Cantidad de Hilos: Esta tabla ayuda a comparar los tiempos promedio en cada máquina y tipo de multiplicación a medida que se incrementa el número de hilos

Máquina	Tipo de Multiplicación	Tamaño de Matriz	1 Hilo (us)	2 Hilos (us)	4 Hilos (us)
Máquina 1	Clásica	800x800	1959843.25	1440118	906451.25
Máquina 1	Clásica	1000x1000	3611311	1931048.75	1524883.5
Máquina 1	Clásica	1800x1800	23579459.8	11999600.75	6053092.5
Máquina 1	Transpuesta	800x800	1835058.5	1083716.75	819452.25
Máquina 1	Transpuesta	1000x1000	3070958.25	1780846.5	1066263
Máquina 1	Transpuesta	1800x1800	16075102	8353936.25	4430013.25
Máquina 2	Clásica	800x800	2730084.5	1844034.25	1138328.75
Máquina 2	Clásica	1000x1000	4350691.25	2665978.75	1545597.25
Máquina 2	Clásica	1800x1800	23567651.8	12957971.5	6270353.75
Máquina 2	Transpuesta	800x800	2428610.25	1314904.25	1003419.75
Máquina 2	Transpuesta	1000x1000	3657533.75	2195770.5	1358632.25
Máquina 2	Transpuesta	1800x1800	17474343.3	9302081.75	5344230.5

Figura 27. Tabla de tiempo promedio por máquina. Elaboración propia

Las cuales tienen las siguientes graficas resultantes presentadas en la figura 28 y 29:

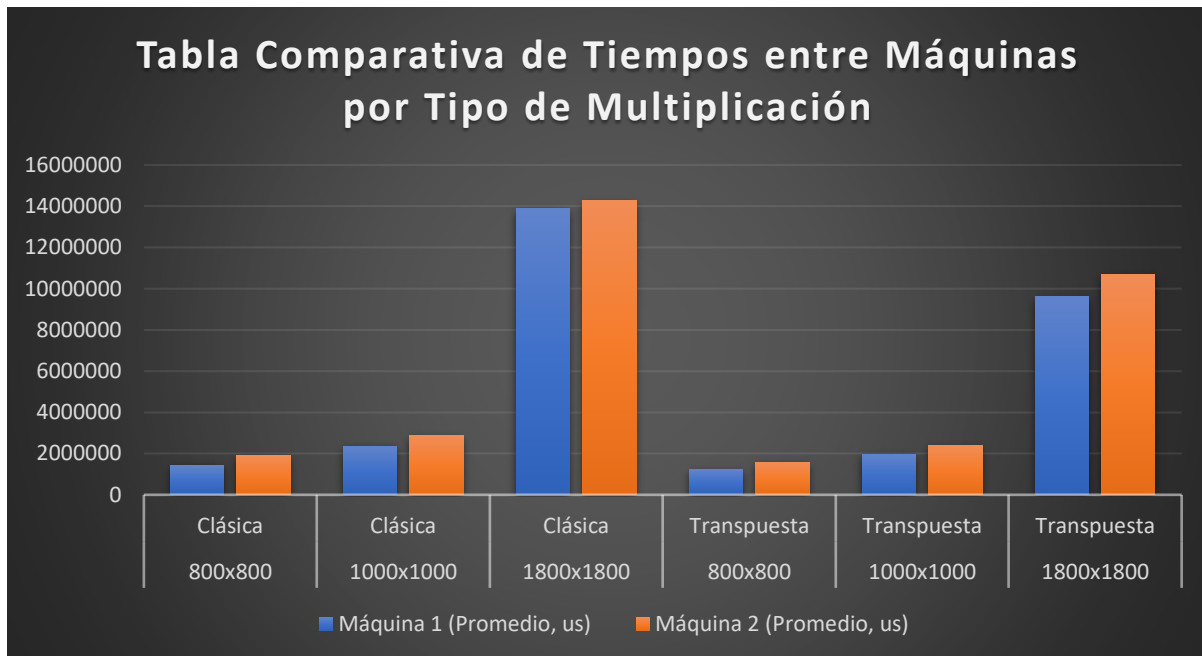


Figura 28. Tabla de tiempo promedio por máquina. Elaboración propia

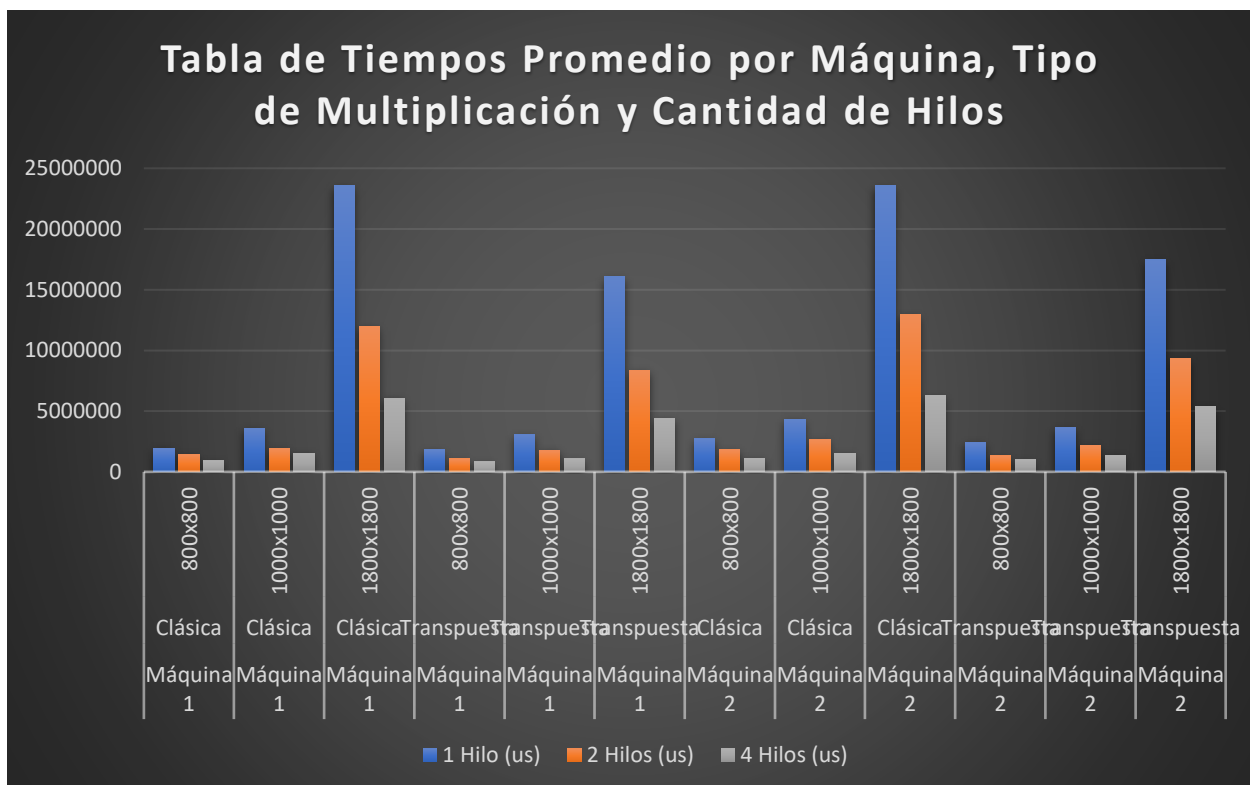


Figura 29. Tabla de tiempo promedio por máquina. Elaboración propia

De las cuales como se puede observar en la figura 28, los tiempos de ejecución de la maquina 1 en promedio son menores que los de la maquina 2, sin embargo esto es un dato muy curioso pues al comparar las jerarquías de memoria en las figuras 9 y 10, junto al primer análisis de la tabla inicial indicaban que la maquina 2 tenía un mejor rendimiento. Sin embargo, en contraste con esta afirmación, en la figura 29 podemos observar que a mayor número de hilos sin importar la maquina ni el tamaño de la matriz los tiempos de ejecución se reducen. Por lo que en conclusión, en general, la **Máquina 2** muestra un mejor rendimiento en ambas configuraciones (clásica y transpuesta) que **Máquina 1**, con tiempos más cortos especialmente para el caso de 4 hilos pues, además, la multiplicación transpuesta en **Máquina 2** es particularmente rápida en comparación con la clásica, indicando una optimización significativa al utilizar la transposición para matrices grandes.