**Cuenta de eventos:**

Terminal : **perf stat -r 5 ./matrix**

**Result: 127840000.000000**

**Result: 127840000.000000**

**Result: 127840000.000000**

**Result: 127840000.000000**

**Result: 127840000.000000**

**Performance counter stats for './matrix' (5 runs):**

**4334,283862 task-clock # 0,998 CPUs utilized ( +- 0,69% )**

**28 context-switches # 0,000 M/sec ( +- 16,58% )**

**0 CPU-migrations # 0,000 M/sec**

**3.879 page-faults # 0,001 M/sec ( +- 0,01% )**

**<not supported> cycles**

**<not supported> stalled-cycles-frontend**

**<not supported> stalled-cycles-backend**

**<not supported> instructions**

**<not supported> branches**

**<not supported> branch-misses**

**4,342234570 seconds time elapsed ( +- 0,70% )**

Terminal: **perf stat -r 5 ./matrix2**

**Result: 127840000.000000**

**Result: 127840000.000000**

**Result: 127840000.000000**

**Result: 127840000.000000**

**Result: 127840000.000000**

**Performance counter stats for './matrix2' (5 runs):**

**1133,233485 task-clock # 0,986 CPUs utilized ( +- 0,53% )**

**27 context-switches # 0,000 M/sec ( +- 11,59% )**

**0 CPU-migrations # 0,000 M/sec**

**5.130 page-faults # 0,005 M/sec ( +- 0,00% )**

**<not supported> cycles**

**<not supported> stalled-cycles-frontend**

**<not supported> stalled-cycles-backend**

**<not supported> instructions**

**<not supported> branches**

**<not supported> branch-misses**

**1,148809898 seconds time elapsed ( +- 0,49% )**

Podemos observar una diferencia de tiempo de ejecución de los programas, siendo matrix2 mas rápido que matrix1. Además, matrix2 tiene más fallos de página que matrix1.

**Terminal : perf stat -e L1-dcache-loads,L1-dcache-load-misses,L1-dcache-stores,L1-dcache-store-misses ./matrix**

**Result: 127840000.000000**

**Performance counter stats for './matrix':**

**<not supported> L1-dcache-loads**

**<not supported> L1-dcache-load-misses**

**<not supported> L1-dcache-stores**

**<not supported> L1-dcache-store-misses**

**1,275606566 seconds time elapsed**

Al ser ejecutado en la máquina virtual y no poder instalar los paquetes en local por no tener permisos, no podemos obtener la información de la caché L1.

Terminal 1: **perf record ./edges img.pgm out.pgm**

Terminal 2: **perf report --stdio**

**captured on: Sun Mar 10 16:56:41 2019**

**# hostname : debian**

**# os release : 3.2.0-4-amd64**

**# perf version : 3.2.101**

**# arch : x86\_64**

**# nrcpus online : 3**

**# nrcpus avail : 3**

**# cpudesc : Intel(R) Core(TM) i7-5700HQ CPU @ 2.70GHz**

**# cpuid : GenuineIntel,6,71,1**

**# total memory : 1026776 kB**

**# cmdline : /usr/bin/perf\_3.2 record ./edges img.pgm out.pgm**

**# event : name = cycles, type = 1, config = 0x0, config1 = 0x0, config2 = 0x0, excl\_usr = 0,**

**# HEADER\_CPU\_TOPOLOGY info available, use -I to display**

**# HEADER\_NUMA\_TOPOLOGY info available, use -I to display**

**# ========**

**#**

**# Events: 5K cpu-clock**

**#**

**# Overhead Command Shared Object Symbol**

**# ........ ....... ................. ..........................**

**#**

**69.03% edges edges [.] gaussian**

**23.71% edges edges [.] laplacian**

**2.34% edges libc-2.13.so [.] fputc**

**2.14% edges libc-2.13.so [.] \_IO\_getc**

**1.08% edges edges [.] load\_image\_file**

**0.79% edges edges [.] save\_image\_file**

**0.09% edges [kernel.kallsyms] [k] native\_read\_tsc**

**0.09% edges [kernel.kallsyms] [k] arch\_local\_irq\_restore**

**0.07% edges edges [.] fputc@plt**

**0.07% edges [kernel.kallsyms] [k] arch\_local\_irq\_enable**

**0.07% edges [kernel.kallsyms] [k] copy\_user\_generic\_string**

**0.05% edges [kernel.kallsyms] [k] arch\_local\_irq\_restore**

**0.05% edges [kernel.kallsyms] [k] dput**

**0.04% edges [kernel.kallsyms] [k] do\_raw\_spin\_lock**

**0.04% edges [kernel.kallsyms] [k] \_cond\_resched**

**0.02% edges edges [.] fgetc@plt**

La función gaussian y laplacian son las que ocupan más tiempo en el programa (69.03 y 23.71% respectivamente).

Terminal 1 :**perf record -g ./edges img.pgm out.pgm**

Terminal 2: **perf report --stdio**

**# captured on: Sun Mar 10 17:00:44 2019**

**# hostname : debian**

**# os release : 3.2.0-4-amd64**

**# perf version : 3.2.101**

**# arch : x86\_64**

**# nrcpus online : 3**

**# nrcpus avail : 3**

**# cpudesc : Intel(R) Core(TM) i7-5700HQ CPU @ 2.70GHz**

**# cpuid : GenuineIntel,6,71,1**

**# total memory : 1026776 kB**

**# cmdline : /usr/bin/perf\_3.2 record -g ./edges img.pgm out.pgm**

**# event : name = cycles, type = 1, config = 0x0, config1 = 0x0, config2 = 0x0, excl\_usr = 0,**

**# HEADER\_CPU\_TOPOLOGY info available, use -I to display**

**# HEADER\_NUMA\_TOPOLOGY info available, use -I to display**

**# ========**

**#**

**# Events: 5K cpu-clock**

**#**

**# Overhead Command Shared Object Symbol**

**# ........ ....... ................. ....................................**

**#**

**69.08% edges edges [.] gaussian**

**|**

**--- gaussian**

**edges**

**main**

**\_\_libc\_start\_main**

**23.46% edges edges [.] laplacian**

**|**

**--- laplacian**

**edges**

**main**

**\_\_libc\_start\_main**

**2.31% edges libc-2.13.so [.] fputc**

La opción -g muestra el árbol de llamadas

Terminal 1: **perf record -e page-faults ./edges img.pgm out.pgm**

Terminal 2: **perf report --stdio**

**# captured on: Sun Mar 10 17:07:20 2019**

**# hostname : debian**

**# os release : 3.2.0-4-amd64**

**# perf version : 3.2.101**

**# arch : x86\_64**

**# nrcpus online : 3**

**# nrcpus avail : 3**

**# cpudesc : Intel(R) Core(TM) i7-5700HQ CPU @ 2.70GHz**

**# cpuid : GenuineIntel,6,71,1**

**# total memory : 1026776 kB**

**# cmdline : /usr/bin/perf\_3.2 record -e page-faults ./edges img.pgm out.pgm**

**# event : name = page-faults, type = 1, config = 0x2, config1 = 0x0, config2 = 0x0, excl\_usr**

**# HEADER\_CPU\_TOPOLOGY info available, use -I to display**

**# HEADER\_NUMA\_TOPOLOGY info available, use -I to display**

**# ========**

**#**

**# Events: 34 page-faults**

**#**

**# Overhead Command Shared Object Symbol**

**# ........ ....... ................. ...............**

**#**

**57.47% edges edges [.] gaussian**

**36.81% edges edges [.] load\_image\_file**

**5.66% edges ld-2.13.so [.] 0xaf0**

**0.02% edges libc-2.13.so [.] 0x7f0b0**

**0.02% edges [kernel.kallsyms] [k] load\_elf\_binary**

**0.02% edges [kernel.kallsyms] [k] \_\_clear\_user**

La opción -e muestra eventos, en este caso con *-e page-faults* muestra los fallos de página(34).

Terminal 1: **perf record -F 250 page-faults ./edges img.pgm out.pgm**

Terminal 2: **perf report --stdio**

**# Events: 1K cpu-clock**

**#**

**# Overhead Command Shared Object Symbol**

**# ........ ....... ................. ...........................**

**#**

**68.60% edges edges [.] gaussian**

**24.03% edges edges [.] laplacian**

**2.18% edges libc-2.13.so [.] fputc**

**2.12% edges libc-2.13.so [.] \_IO\_getc**

**1.06% edges edges [.] save\_image\_file**

**1.00% edges edges [.] load\_image\_file**

**0.25% edges [kernel.kallsyms] [k] arch\_local\_irq\_restore**

**0.06% edges edges [.] fgetc@plt**

**0.06% edges edges [.] fputc@plt**

**0.06% edges [kernel.kallsyms] [k] native\_read\_tsc**

**0.06% edges [kernel.kallsyms] [k] arch\_local\_irq\_enable**

**0.06% edges [kernel.kallsyms] [k] need\_resched**

**0.06% edges [kernel.kallsyms] [k] generic\_file\_buffered\_write**

**0.06% edges [kernel.kallsyms] [k] mark\_page\_accessed**

**0.06% edges [kernel.kallsyms] [k] arch\_local\_irq\_restore**

**0.06% edges [kernel.kallsyms] [k] touch\_atime**

La opción -F indica la frecuencia en Hz con la que se toman muestras. A mayor número (por ejemplo -F 250) mayor es el detalle de la información.

**usuario@debian:~/Documentos$ perf record -c 1000 ./edges img.pgm out.pgm**

**[ perf record: Woken up 3 times to write data ]**

**[ perf record: Captured and wrote 0.584 MB perf.data (~25510 samples) ]**

**usuario@debian:~/Documentos$ perf record -c 100 ./edges img.pgm out.pgm**

**[ perf record: Woken up 3 times to write data ]**

**[ perf record: Captured and wrote 0.627 MB perf.data (~27387 samples) ]**

# Events: 19K cpu-clock

#

# Overhead Command Shared Object Symbol

# ........ ....... ................. ..................................

#

61.41% edges edges [.] gaussian

17.69% edges edges [.] laplacian

8.12% edges [kernel.kallsyms] [k] arch\_local\_irq\_restore

6.22% edges [kernel.kallsyms] [k] arch\_local\_irq\_enable

2.41% edges libc-2.13.so [.] fputc

1.38% edges libc-2.13.so [.] \_IO\_getc

0.90% edges edges [.] save\_image\_file

0.77% edges edges [.] load\_image\_file

0.21% edges [kernel.kallsyms] [k] arch\_local\_irq\_restore

0.16% edges [kernel.kallsyms] [k] native\_read\_tsc

0.08% edges edges [.] fputc@plt

0.07% edges [scsi\_mod] [k] spin\_unlock\_irq

0.07% edges edges [.] fgetc@plt

La opción -c indica el periodo con el que se tienen que obtener las muestras.

gprof:

**usuario@debian:~/Documentos$ gcc -O0 -pg edges.c -o edges\_gprof**

**usuario@debian:~/Documentos$ ./edges\_gprof img.pgm out2.pgm**

**usuario@debian:~/Documentos$ gprof edges\_gprof gmon.out > info.txt**

**usuario@debian:~/Documentos$ nano info.txt**

Each sample counts as 0.01 seconds.

% cumulative self self total

time seconds seconds calls s/call s/call name

76.20 4.19 4.19 2 2.10 2.10 gaussian

21.51 5.37 1.18 1 1.18 1.18 laplacian

1.64 5.46 0.09 1 0.09 0.09 load\_image\_file

0.91 5.51 0.05 1 0.05 0.05 save\_image\_file

0.00 5.51 0.00 1 0.00 5.37 edges

El comando perf nos muestra la secuencia de eventos que se realizan cuando se ejecuta un programa.

El comando gprof nos permite hacer un análisis de los tiempos de las diferentes funciones de un programa.

Google-pprof crea un árbol de llamadas del programa mediante un grafo.

**Valgrind:**

Terminal:  **valgrind --tool=cachegrind ./matrix**

==4447==

==4447== I refs: 261,973,485

==4447== I1 misses: 672

==4447== LLi misses: 661

==4447== I1 miss rate: 0.00%

==4447== LLi miss rate: 0.00%

==4447==

==4447== D refs: 85,607,963 (76,506,421 rd + 9,101,542 wr)

==4447== D1 misses: 8,217,147 ( 8,046,943 rd + 170,204 wr)

==4447== LLd misses: 164,016 ( 2,348 rd + 161,668 wr)

==4447== D1 miss rate: 9.5% ( 10.5% + 1.8% )

==4447== LLd miss rate: 0.1% ( 0.0% + 1.7% )

==4447==

==4447== LL refs: 8,217,819 ( 8,047,615 rd + 170,204 wr)

==4447== LL misses: 164,677 ( 3,009 rd + 161,668 wr)

==4447== LL miss rate: 0.0% ( 0.0% + 1.7% )

Terminal: valgrind --tool=cachegrind ./matrix2

==4427==

==4427== I refs: 581,282,073

==4427== I1 misses: 675

==4427== LLi misses: 664

==4427== I1 miss rate: 0.00%

==4427== LLi miss rate: 0.00%

==4427==

==4427== D refs: 188,181,121 (168,668,218 rd + 19,512,903 wr)

==4427== D1 misses: 3,076,447 ( 2,273,229 rd + 803,218 wr)

==4427== LLd misses: 288,584 ( 44,594 rd + 243,990 wr)

==4427== D1 miss rate: 1.6% ( 1.3% + 4.1% )

==4427== LLd miss rate: 0.1% ( 0.0% + 1.2% )

==4427==

==4427== LL refs: 3,077,122 ( 2,273,904 rd + 803,218 wr)

==4427== LL misses: 289,248 ( 45,258 rd + 243,990 wr)

==4427== LL miss rate: 0.0% ( 0.0% + 1.2% )

Se observa que valgrind muestra información más detallada que perf (fallos lectura y escritura). Además, matrix2 tiene menos fallos de caché que matrix1.

strace:

Terminal: **sudo strace -e open vmstat**

open("/etc/ld.so.cache", O\_RDONLY) = 3

open("/lib/x86\_64-linux-gnu/libprocps.so.0", O\_RDONLY) = 3

open("/lib/x86\_64-linux-gnu/libc.so.6", O\_RDONLY) = 3

open("/proc/stat", O\_RDONLY|O\_CLOEXEC) = 3

open("/usr/lib/locale/locale-archive", O\_RDONLY) = 3

open("/usr/share/locale/locale.alias", O\_RDONLY) = 3

open("/usr/share/locale/es\_ES.UTF-8/LC\_MESSAGES/procps-ng.mo", O\_RDONLY) = -1 ENOENT (No such file or directory)

open("/usr/share/locale/es\_ES.utf8/LC\_MESSAGES/procps-ng.mo", O\_RDONLY) = -1 ENOENT (No such file or directory)

open("/usr/share/locale/es\_ES/LC\_MESSAGES/procps-ng.mo", O\_RDONLY) = -1 ENOENT (No such file or directory)

open("/usr/share/locale/es.UTF-8/LC\_MESSAGES/procps-ng.mo", O\_RDONLY) = -1 ENOENT (No such file or directory)

open("/usr/share/locale/es.utf8/LC\_MESSAGES/procps-ng.mo", O\_RDONLY) = -1 ENOENT (No such file or directory)

open("/usr/share/locale/es/LC\_MESSAGES/procps-ng.mo", O\_RDONLY) = -1 ENOENT (No such file or directory)

procs -----------memory---------- ---swap-- -----io---- -system-- ----cpu----

r b swpd free buff cache si so bi bo in cs us sy id wa

open("/proc/meminfo", O\_RDONLY) = 3

open("/proc/stat", O\_RDONLY) = 4

open("/proc/vmstat", O\_RDONLY) = 5

0 0 92 103472 12568 588120 0 0 122 322 57 301 3 1 95 1

Obtiene la información de la carpeta /proc, además otras carpetas internas como /proc/meminfo.

Terminal: **sudo strace -c -o fichero.txt find /usr &> /dev/null**

time seconds usecs/call calls errors syscall

------ ----------- ----------- --------- --------- ----------------

92.42 0.000061 0 6298 getdents

3.03 0.000002 0 6306 close

3.03 0.000002 0 3156 fstat

1.52 0.000001 0 6310 5 open

0.00 0.000000 0 6 read

0.00 0.000000 0 462 write

0.00 0.000000 0 20 mmap

0.00 0.000000 0 10 mprotect

0.00 0.000000 0 3 munmap

0.00 0.000000 0 30 brk

0.00 0.000000 0 2 rt\_sigaction

0.00 0.000000 0 1 rt\_sigprocmask

0.00 0.000000 0 3 2 ioctl

0.00 0.000000 0 6 6 access

0.00 0.000000 0 1 execve

0.00 0.000000 0 1 uname

0.00 0.000000 0 6299 fchdir

0.00 0.000000 0 1 getrlimit

0.00 0.000000 0 1 arch\_prctl

0.00 0.000000 0 1 1 futex

0.00 0.000000 0 1 set\_tid\_address

0.00 0.000000 0 3148 newfstatat

0.00 0.000000 0 1 set\_robust\_list

------ ----------- ----------- --------- --------- ----------------

100.00 0.000066 32067 14 total

**getdents ()**

lee varias estructuras linux\_dirent desde el directorio al que hace referencia el descriptor de archivo abierto fd en el buffer apuntado por dirp .

**close ()**

cierra un descriptor de archivo, de modo que ya no hace referencia a ningún archivo y puede reutilizarse.

**fstat ()**

es idéntica a stat (), que devuelve las estadísticas del archivo apuntado por ruta y rellena buf ,excepto que en fstat() el archivo stat-ed es especificado por el descriptor de fichero fd .

**open ()**

establece la conexión entre un archivo y un descriptor de archivo.

**read ()**

intenta leer el número de bytes del archivo descriptor fd en el búfer a partir de otro búfer indicado.