
ACTIVIDAD SYSBENCH

ALUMNO: ADRIÁN BENNASAR POLZIN – GRUPO MAÑANA

Ejecutar el benchmark Sysbench CPU para el máximo número de CPUs de la máquina disponible.

- **¿Cuánto tiempo tarda en ejecutarse? (hacer uso del monitor time)**

El benchmark se ha ejecutado con 4 CPUs, que es el máximo de mi máquina. Le he puesto un número de primos para que tarde 2 minutos más o menos (como se indica en el vídeo de la profesora), en este caso el número de primos ha sido 50000.

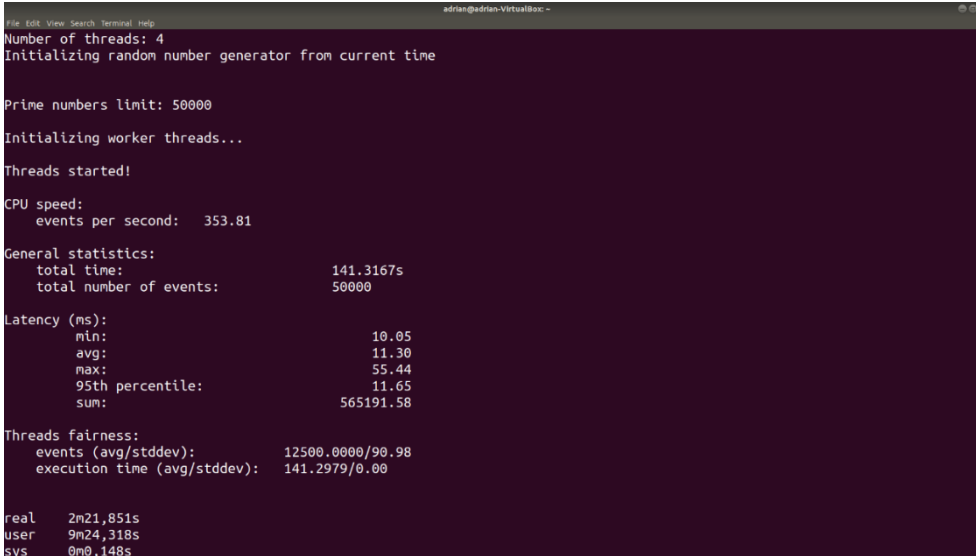
Con estos parámetros, el monitor time ejecutado sobre el benchmark, ha devuelto un tiempo de **141.851** segundos.

El comando completo utilizado es:

```
time sysbench --test=cpu --cpu-max-prime=50000 --threads=4 --events=50000 --time=0 run
```

- **¿Coincide con el tiempo proporcionado por el benchmark?**

El benchmark ha devuelto un tiempo de **141.3167** segundos. Es decir, hay una pequeña diferencia entre el tiempo del monitor time y el del benchmark. Esta diferencia puede deberse a que el monitor time cuenta el tiempo que necesita el benchmark para procesar los parámetros y arrancar con dicha configuración, mientras que el tiempo del benchmark solo empieza a contar una vez que el benchmark ya ha arrancado.



```
adrian@adrian-VirtualBox: ~
File Edit View Search Terminal Help
Number of threads: 4
Initializing random number generator from current time

Prime numbers limit: 50000
Initializing worker threads...
Threads started!

CPU speed:
  events per second: 353.81

General statistics:
  total time:          141.3167s
  total number of events: 50000

Latency (ms):
  min:                10.05
  avg:                 11.30
  max:                 55.44
  95th percentile:    11.65
  sum:                 565191.58

Threads fairness:
  events (avg/stddev): 12500.0000/90.98
  execution time (avg/stddev): 141.2979/0.00

real    2m21.851s
user    9m24.318s
sys     0m0.148s
```

- **Realizar dicha ejecución para 2 CPUs, 4 CPUs, ..., hasta el número máximo permitido. ¿Cuál ha sido la utilización media de la CPU en cada caso?**

Para esta parte de la actividad, el método utilizado ha sido hacer una prueba piloto para cada número de threads y ver el tiempo que tarda en ejecutarse la carga, y entonces volver a ejecutar dicha carga para los 4 casos, ejecutando también el monitor top en otra terminal, ajustando los parámetros de número de muestras y tiempo entre muestras para que se ejecute durante aproximadamente el mismo tiempo que el benchmark.

Si el monitor se ha pasado de unos pocos segundos, lo que he hecho es quitar 1 o 2 muestras, para que coincida con el benchmark.

El output del monitor top lo he redireccionado a un fichero, y para calcular la utilización media en global de la CPU, al igual que como hice en la práctica de monitorización, he hecho la media del cálculo **100 – iddle**.

Resultados:

1 thread

- Utilización global teórica: **25%**
- Utilización global obtenida: **25.46%**

2 threads

- Utilización global teórica: **50%**
- Utilización global obtenida: **50.24%**

3 threads

- Utilización global teórica: **75%**
- Utilización global obtenida: **75.29%**

4 threads

- Utilización global teórica: **100%**
- Utilización global obtenida: **100%**

Como se puede ver, la utilización obtenida se aproxima mucho a la teórica, pero no es exactamente la misma. Esto se debe a que además de la ejecución de la carga, en la máquina hay otros procesos ejecutándose, además del propio sistema operativo, y esto consume algo de CPU.

A continuación, dejo unos gráficos comparativos:

