



Consumidor regulable de CPU



Elaborada por: M. en C. Ukranio Coronilla

Para probar algunas arquitecturas en la nube necesitaremos una aplicación que pueda hacer uso del porcentaje de CPU que se desee por lo que en esta práctica nos daremos a la tarea de programarlo.

Ejercicio 1

Primero elaboraremos un programa en Java que reciba en la línea de comandos un entero el cual representa el porcentaje de CPU (entre 1% y 100%) que se desea utilizar, y el tiempo en segundos durante el cual se va a estar ejecutando el programa. El porcentaje de uso del CPU lo podremos visualizar en una terminal UNIX mediante el comando **htop**, por lo que al terminar su programa, deberá enviar además de su código, la captura de las **dos** terminales donde se muestra que funciona correctamente, por ejemplo:

The screenshot shows two terminal windows. The top window displays the output of the `htop` command, showing system statistics and a list of processes. A red box highlights the CPU usage bar, which shows 62.0% usage. The bottom window shows the execution of the `java SimulaCargaCpu 65 10` command, which simulates 65% CPU usage for 10 seconds.

```
ukranio@LAPTOP-0CG5G04:~$ htop
 0[ | 2.6% 4[ | 62.0%
 1[ | 2.0% 5[ | 0.0%
 2[ | 1.3% 6[ | 0.7%
 3[ | 0.0% 7[ | 0.0%
Mem[ | 541M/3.35G Tasks: 38, 62 thr; 2 running
Swp[ | 0K/1.00G Load average: 0.07 0.05 0.01
Uptime: 00:08:58

  PID USER   PRI  NI  VIRT   RES   SHR  S CPU% MEM%   TIME+  Command
 1035 ukranio   20    0 3604M 41872 20728 S 64.1  1.2  0:02.49 java SimulaCargaCpu 65 10

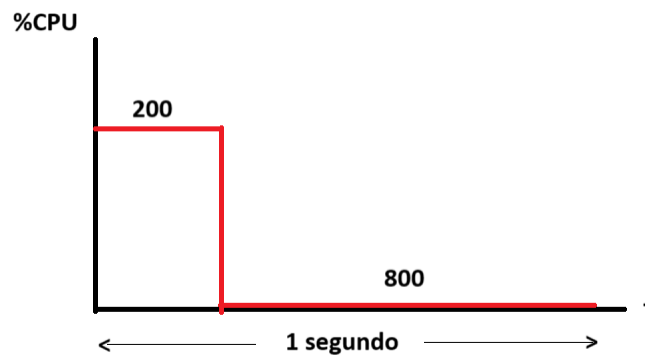
ukranio@LAPTOP-0CG5G048:~/SD_2024_AGO8TO/Servidor_CPU$ time java SimulaCargaCpu 65 10
Se va a simular un uso de CPU del 65% durante 10 segundos.
```

Observe que el programa debe imprimir el porcentaje de uso de CPU y el tiempo durante el cual se va a ejecutar tal y como se recibió en los argumentos del programa.

A continuación, se explica cómo sería posible hacer este programa.

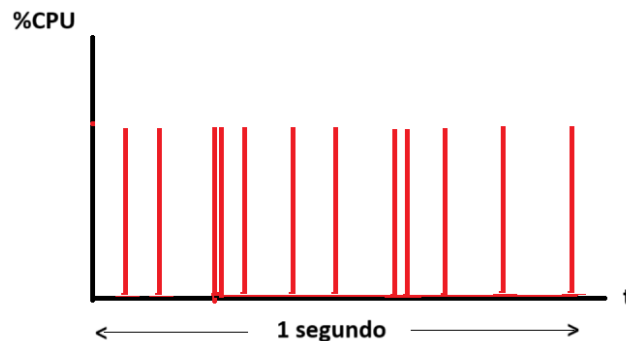
Suponga que deseamos utilizar un 20% del CPU, dado que en realidad un CPU está en funcionamiento o no lo está, no podemos ejecutar operaciones que sólo utilizan

un porcentaje de este. Por lo tanto, tendremos que hacer uso del CPU un 20% del tiempo y dejarlo de usar un 80% del tiempo. Para lograrlo dividimos un segundo de tiempo en intervalos de 1 milisegundo por lo que en un segundo tendremos mil intervalos de tiempo. Así si queremos simular un uso del 20% del CPU entonces en doscientos intervalos se utilizará el CPU y en ochocientos no se utilizará. Ahora si los intervalos de uso de CPU los ponemos juntos tendremos algo como:



Lo cual provocará que en el visualizador de htop se dispare al 100% el uso de CPU durante las primeras dos décimas de segundo y baje a 0% en las últimas ocho décimas de segundo.

Para que esto no suceda tendremos que distribuir de manera aleatoria los 200 intervalos de uso de CPU en el periodo de un segundo:



De esta manera el monitor de uso de CPU percibirá un uso del 20% al obtener un promedio sobre el tiempo.

Utilizaremos la siguiente línea de código que obtiene la raíz cuadrada de un número aleatorio para hacer trabajar al CPU:

```
Math.sqrt(ran.nextInt(2147483647));
```

Y esta línea se deberá ejecutar en un ciclo que valida constantemente si ha transcurrido el milisegundo al compararlo con el tiempo inicial obtenido con la función:

```
System.currentTimeMillis()
```

Ejercicio 2

Para probar el programa en CPUs multinúcleo reciba adicionalmente en la línea de comandos un tercer número entero *n* que indica el número de hilos que van a ejecutar el código del ejercicio anterior. Mande una captura de pantalla similar a la anterior para probar que efectivamente funciona para *n* núcleos.

Ejercicio 3

Incorpore el endpoint **/cpu** en un servidor HTTP de manera que ejecute el código del ejercicio anterior recibiendo como datos del mensaje el porcentaje de CPU, el tiempo en segundos y el número de núcleos. Probarlo con curl enviando la captura de pantalla de su ejecución junto con la ejecución del comando htop.