

Multiprocesamiento y arquitecturas alternativas

Proyecto 1er bimestre

Título: Pruebas y comparaciones de programas multihilo.

Nombre: Luis Enrique Pérez Señalín.

Informe:

1. INTRODUCCIÓN:

Los computadores actuales cuentan con múltiples hilos, también conocidos como “procesadores lógicos” los cuales son una división “lógica” de un núcleo de procesador, lo que significa que de un núcleo físico pueden sacar 1 o más núcleos lógicos.

Los procesadores actuales cuentan con una gran cantidad de hilos, pero no todos los programas utilizan todos los hilos para funcionar, lo que en ciertos casos es un ahorro de consumo de recursos y en otros un desperdicio de recursos por falta de uso.

En este proyecto vamos a comparar las velocidades de procesamiento entre varios procesadores, que son de distintos fabricantes, generación y arquitectura, así como sus gamas y la cantidad de núcleos e hilos que poseen.

Vamos a utilizar el lenguaje Java implementando un algoritmo para el calculo del número de Fibonacci, el cual contará con la capacidad de elegir el número de hilos y el número de cálculos que se harán.

2. DESARROLLO:

2.1 Descripción del programa.

El código del proyecto cuenta con dos partes, la principal es la llamada al algoritmo para el cálculo del número de Fibonacci que se hará utilizando multihilos en java y el algoritmo recursivo para el código de Fibonacci.

2.2 Descripción del programa principal.

El algoritmo principal funciona utilizando un grupo de números que se van a utilizar para calcular su número de Fibonacci, con el fin de distribuir estos cálculos a los demás hilos del procesador, significa que podemos tener 5 cálculos del número 45 de Fibonacci. Utilizamos de 1 a n cantidad de hilos dentro del programa, la máxima especificación solo permitirá la cantidad de hilos que tenga el equipo, si este se excede solo usará la cantidad que tiene.

2.3 Descripción del programa secundario.

Para realizar el cálculo del número de Fibonacci implementamos una función recursiva de la forma $F_n = F_{n-1} + F_{n-2}$, con la condición de cuando $n \leq 1$ retorne el valor de n.

2.4 Parámetros de entrada del programa.

Tendremos de entrada 3 parámetros por medio de los argumentos del sistema, significa que se ingresarán al momento de ejecutar el comando de inicialización del programa, en concreto tenemos este orden:

1. El número de Fibonacci a calcular, como recomendación números mayores a 45 suelen tardar más 3 segundos en computadores actuales entonces se evitaría usar números muy elevados.
2. La cantidad de veces que se va a calcular el número de Fibonacci ingresado anteriormente.
3. El número de hilos que el programa va a utilizar, el cuál debe ser mínimo 1, si se ingresa más hilos de los que el equipo tiene, entonces solo se usarán el número de hilos máximos que tenga.

2.5 Utilización de múltiples hilos.

Para hacer uso de múltiples hilos en nuestro programa, usamos la librería de “ExecutorService” para crear un objeto “executor” que importamos de esta manera:

```
import java.util.concurrent.ExecutorService;  
import java.util.concurrent.Executors;
```

La librería “ExecutorService” y “executor” nos permitirá crear un objeto de la siguiente manera:

```
ExecutorService executor = Executors.newFixedThreadPool(threads);
```

Para utilizar el objeto “executor” podemos hacerlo en un ciclo repetitivo cualquiera o asignando manualmente un proceso de la siguiente manera:

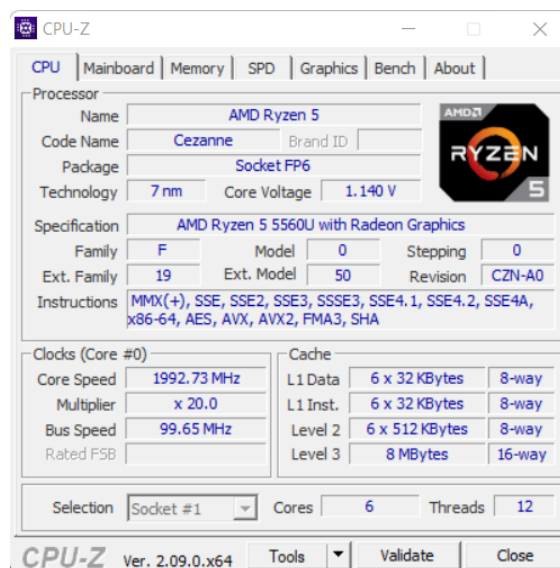
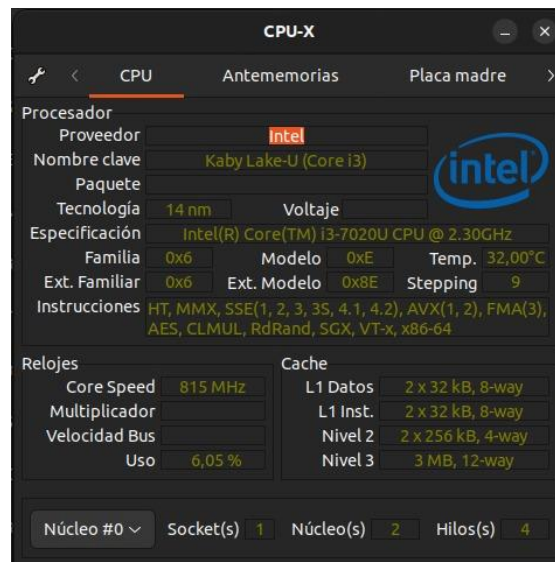
```
executor.submit(() -> { fibonacci(number) });
```

De esta forma le asignamos un proceso al objeto executor.

2.6 Descripción de los equipos de cómputo.

Para estas pruebas se tomaron 3 diferentes equipos, de los cuales 2 tienen procesadores de tipo laptop y 1 de tipo escritorio.

- Equipo: Laptop marca Dell
Procesador: Intel I3-7020U – 2 núcleos 4 hilos
Memoria: 8gb ram ddr4 2400Mhz
SO: Ubuntu 22.04 LTS.
- Equipo: Minipc Beelink SER5
Procesador: Ryzen 5 5560U – 6 núcleos 12 hilos
Memoria: 21gb ram ddr4 3200Mhz
SO: Windows 11.
- Equipo: Pc escritorio
Procesador: Intel I5-1400F – 6 núcleos 12 hilos
Memoria: 16gb ram ddr4 3200Mhz
SO: Windows 10.



Capturas de pantalla de las especificaciones del CPU de los equipos.

2.7 Comparaciones.

Vamos a realizar las comparaciones utilizando una cantidad de 36 cálculos con todos los núcleos disponibles y vamos a ir variando el número de Fibonacci que vamos a calcular.

1. Fibonacci número 44.

- 13-7020U:



Imagen i3-7020U Número 44

- Ryzen 5 5560U:

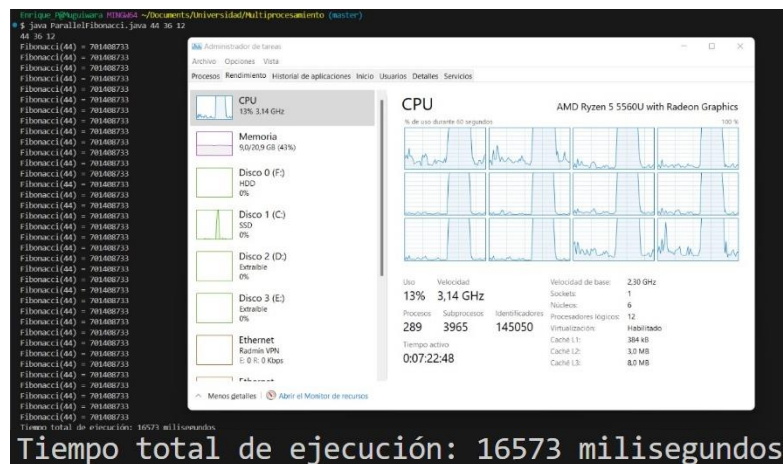


Imagen Ryzen 5 5560U Número 44

- I5-11400F: