



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

**Práctica 2: Ley de Amdahl y Paralelismo**

ALUMNOS

Rivera Zavala Javier Alejandro - 311288876

López Diego Gabriela - 31824385

Diego Martínez Calzada - 318275457

PROFESOR

Gilde Valeria Rodríguez Jiménez

AYUDANTES

Rogelio Alcantar Arenas

Gibran Aguilar Zuñiga

Luis Angel Leyva Castillo

ASIGNATURA

Computación Concurrente

## 1. Actividades

1. Genera una gráfica de tiempo de ejecución vs el número de hilos.



Figura 1: Gráfica de tiempo de ejecución de nuestro código con diferente cantidad de hilos

2. Genera una tabla como la siguiente

Num de Hilos	Aceleración Teórica	Aceleración Obtenida	% Código en Paralelo
1	1	1	0 %
2	1.48235294118	12.61176869	aprox. 65 %
27	2.67874015748	9.05236904	aprox. 65 %
100	2.81124497992	2.34445876	aprox. 65 %

## 2. Cuestionario

1. ¿Cuál fue mi contraseña?

La contraseña fue **"eqpdia"**

2. ¿Cuántas posibles contraseñas hay?

Sabíamos que nuestra contraseña se conformaba de 6 caracteres y nuestro alfabeto de 26 letras, por lo cual el número total de combinaciones posibles o posibles contraseñas era de

$$26^6 = 308915776$$

Esto debido a que puede haber repeticiones de una misma letra en nuestra contraseña (nunca se indico lo contrario) y donde importa el orden.

---

3. ¿La ley del Amdahl siempre se cumple?

No, solo se cumple cuando la fracción de trabajo que puede ser paralelizable permanece fijo.

4. ¿En qué casos no se cumple?

Cuando la carga de trabajo cambia de forma dinámicamente. Si ocurre esto, el calculo de aceleración teórica puede ser incierta e ir cambiando constantemente.

5. ¿A qué crees a que se debe esto?

Esto puede deberse al funcionamiento del procesador y a como este, ante una mayor exigencia de sus capacidades, se vuelve más susceptible de cometer errores o bien se vuelve más lento debido a la liberación de calor generada. La liberación de energía eléctrica transformada en calor, puede conducir al CPU a ejecutar un rutina conocida como regulación del reloj, que limita la velocidad del reloj para evitar daños por calor.

6. ¿Cuál sería la mejora máxima? Es decir, la aceleración teórica máxima.

Dada nuestra implementación podemos apreciar que alrededor del 65 % del código que escribimos es paralelizable por lo tanto, dada la ecuación para el speedrun, tenemos algo como  $s(n) = \frac{1}{(1-0.65) + \frac{0.65}{n}}$ , para conocer la mejora máxima posible, supondremos que disponemos de una cantidad infinita de hilos y por lo tanto hemos de resolver

$$\lim_{n \rightarrow \infty} s(x) = \frac{1}{(1 - 0.65) + \frac{0.65}{n}} = \frac{1}{0.35 + \frac{0.65}{n}}$$

y esto último es igual a  $\frac{1}{\lim_{n \rightarrow \infty} f(n) = 0.35 + \frac{0.65}{n}}$  que es igual a  $\frac{1}{0.35} = 2.857142857$ , lo que corresponde a la aceleración teórica máxima, siguiendo la implementación que entregamos para la solución del problema.

7. Escribe tus conclusiones, además de lo que aprendiste en esta práctica, contratiempos y descubrimientos que hubo durante su realización.

En general fue una práctica sencilla de resolver en el papel, aunque un tanto complicada de correr y finalizar, es decir, diseñar el algoritmo a seguir no nos supuso ningún esfuerzo de consideración, sin embargo, dada la naturaleza de la práctica, obtener los resultados esperados tomo bastante tiempo. Para poder ejecutar la implementación del algoritmo, requerimos que nuestras computadoras se pusieran a trabajar por un buen rato, por ende, cualquier omisión o error en la ejecución que no hubiésemos contemplado desde el principio, nos costó tener que repetir la larga ejecución más

---

de una vez.

Implementamos una solución sencilla e inmediata del problema, no nos complicamos demasiado, ya que no era el propósito de esta práctica desarrollar una solución tan eficiente. Antes que todo, lo que importaba era entender el funcionamiento de la ejecución concurrente/paralela de un programa y el impacto de tal factor en el rendimiento. Por otra parte, dado que intentamos correr nuestro código en las computadoras de los 3 miembros del equipo, mismas que varían bastante en su composición, pudimos notar cambios notables en el rendimiento entre una y otra computadora, pues la arquitectura de cada procesador en particular, le permite un mejor o peor manejo del computo concurrente y paralelo. Nos sorprendió un poco que contrario a nuestras expectativas, un mayor número de hilos no se tradujo en un incremento drástico en el rendimiento real e incluso, dadas las limitaciones de nuestros ordenadores, con pocos hilos obtuvimos un mejor funcionamiento que con muchos de ellos, aunque está por verse si nuestra implementación tiene también algo que ver con ello (será tarea del ayudante informarnos).

8. ¿Cuál es su rol?

Somos hilos a los que se les ha asignado una parte de la tarea a ejecutar.

9. ¿Cuál es mi rol?

Eres el CPU que asigna a cada hilo la carga de trabajo que le corresponde.

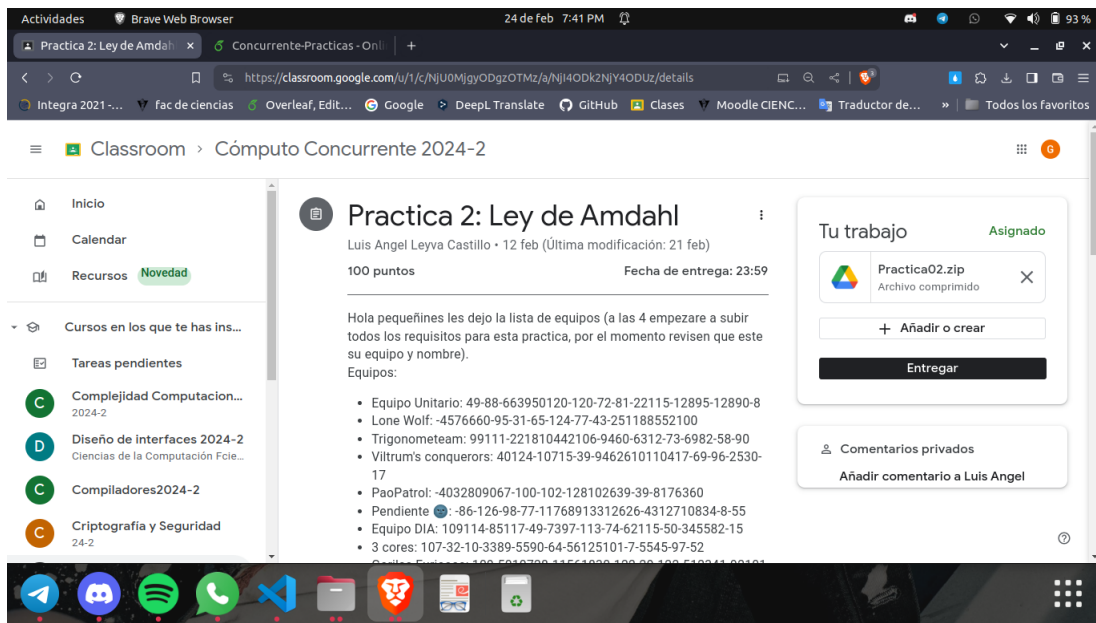


Figura 2: Captura de pantalla cuando agregamos nuestras palabras al documento

### 3. SonarLint

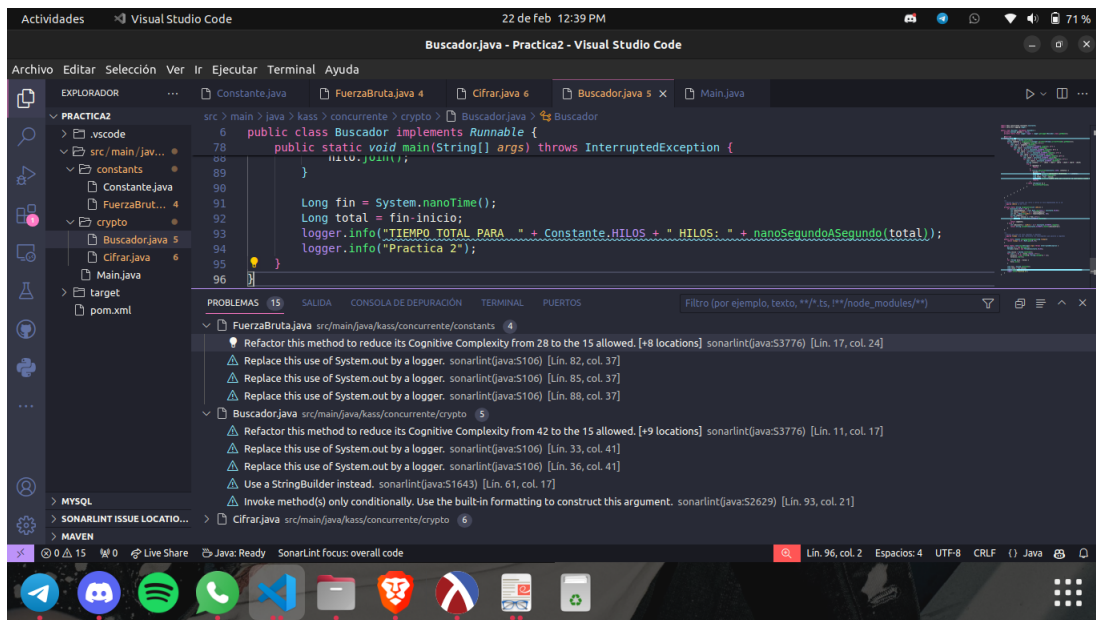


Figura 3: Warnings detectados por SonarLint

---

## 4. Referencias

- La ley de Amdahl. (s/f). Academia-lab.com. Recuperado el 20 de febrero de 2024, de <https://academia-lab.com/enciclopedia/la-ley-de-amdahl/>
- Chad, John. (1 de enero de 2023). Does Cpu Temperature Affect Performance? (Fact Checked). Tech review team (Recuperado el 21 de febrero de 2024).  
<https://techreviewteam.com/hardware/does-cpu-temperature-affect-performance-fact-checked/>