**LABORATORIO 1 ARSW**

Presentado por:

Daniela García

Edwar Lozano

Presentado a:

Diego Triviño

Escuela Colombiana de Ingeniería Julio Garavito

Materia: ARSW

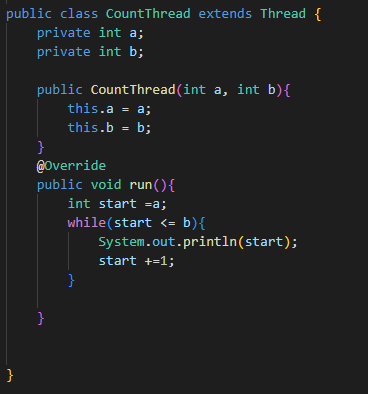
Grupo: 2

1 de febrero de 2023

**LABORATORIO 1 ARSW**

**Parte I Hilos Java**

1. De acuerdo con lo revisado en las lecturas, complete las clases CountThread, para que las mismas definan el ciclo de vida de un hilo que imprima por pantalla los números entre A y B.



1. Complete el método main de la clase CountMainThreads para que:

* Cree 3 hilos de tipo CountThread, asignándole al primero el intervalo [0..99], al segundo [99..199], y al tercero [200..299].

Texto

Descripción generada automáticamente

* Inicie los tres hilos con 'start()'.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente con confianza media

* Ejecute y revise la salida por pantalla.

Imagen que contiene Tabla

Descripción generada automáticamente

* Cambie el inicio con 'start()' por 'run()'. ¿Cómo cambia la salida?, por qué?

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Imagen que contiene Tabla

Descripción generada automáticamente

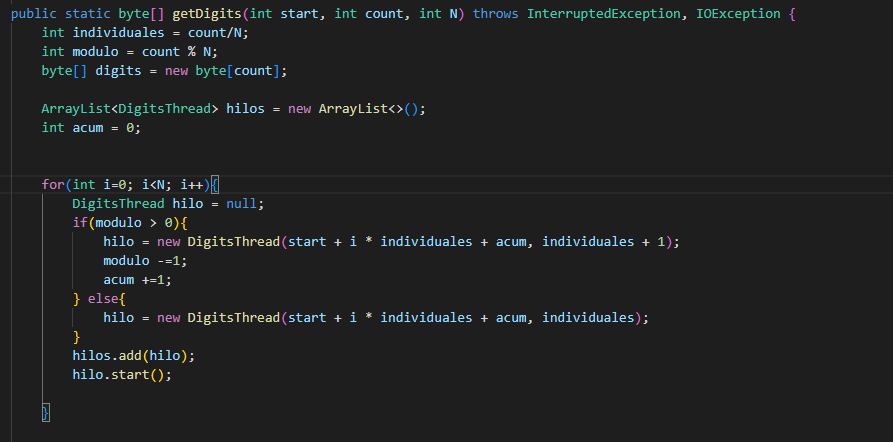
**Parte II Hilos Java**

1. Cree una clase de tipo Thread que represente el ciclo de vida de un hilo que calcule una parte de los dígitos requeridos.

Texto

Descripción generada automáticamente

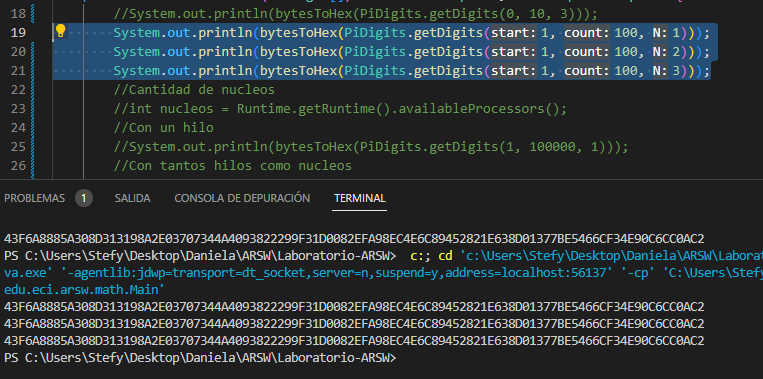
1. Haga que la función PiDigits.getDigits() reciba como parámetro adicional un valor N, correspondiente al número de hilos entre los que se va a paralelizar la solución. Haga que dicha función espere hasta que los N hilos terminen de resolver el problema para combinar las respuestas y entonces retornar el resultado. Para esto, revise el método join del API de concurrencia de Java.



Texto

Descripción generada automáticamente

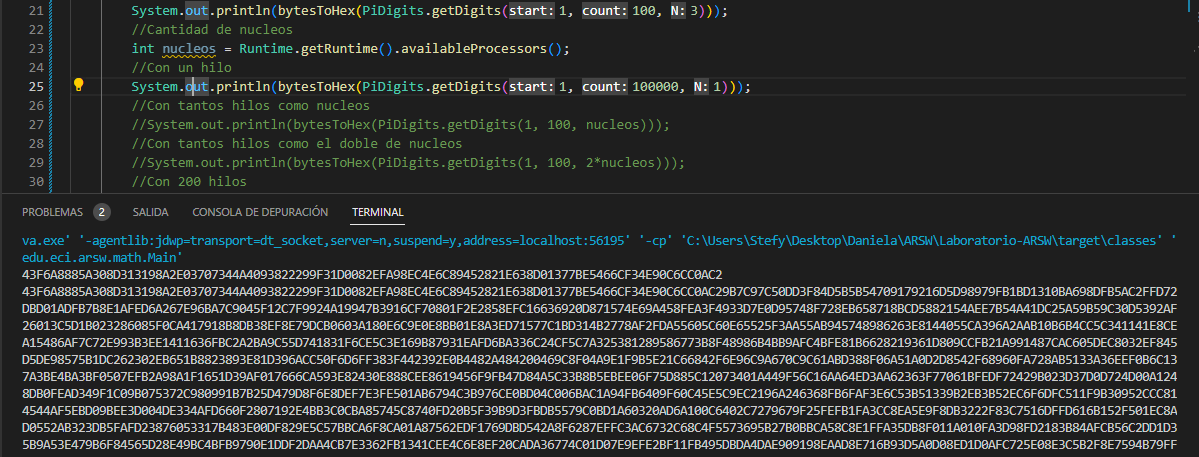
1. Ajuste las pruebas de JUnit, considerando los casos de usar 1, 2 o 3 hilos (¡este último para considerar un número impar de hilos!)



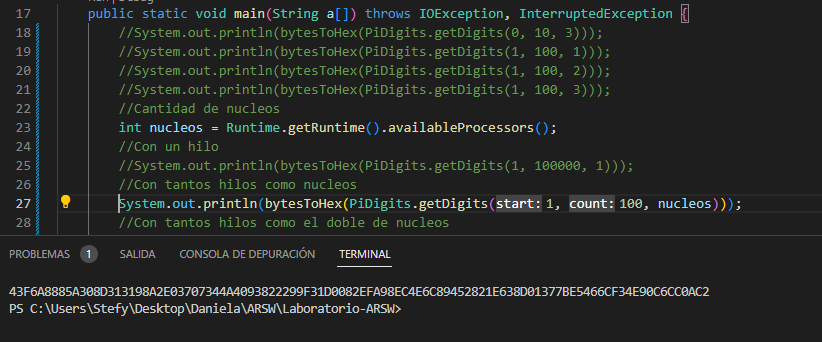
**Parte III Evaluación de Desempeño**

A partir de lo anterior, implemente la siguiente secuencia de experimentos para calcular el millon de dígitos (hex) de PI, tomando los tiempos de ejecución de los mismos (asegúrese de hacerlos en la misma máquina):

1. Un solo hilo.



1. Tantos hilos como núcleos de procesamiento (haga que el programa determine esto haciendo uso del [API Runtime](https://docs.oracle.com/javase/7/docs/api/java/lang/Runtime.html)).



1. Tantos hilos como el doble de núcleos de procesamiento.

Texto

Descripción generada automáticamente

1. 200 hilos.

Texto

Descripción generada automáticamente

1. 500 hilos.

Captura de pantalla de computadora

Descripción generada automáticamente

Al iniciar el programa ejecute el monitor jVisualVM, y a medida que corran las pruebas, revise y anote el consumo de CPU y de memoria en cada caso.