|  |  |  |
| --- | --- | --- |
|  | **UNIVERSIDAD NACIONAL DE JOSE C. PAZ** | |
| **Departamento de Industria e Innovación Tecnológica** | |
| **Carrera:** Analista Programador Universitario | |
| **Asignatura:** Opt1 - **Mesa:** Primer Parcial | |
| **Alumno:** | **Fecha:** 29/04/204 |

### Recomendaciones

Requisitos para aprobar: Para que el parcial esté aprobado deberá tener correctamente desarrollado/ contestado el **60%** del examen.

1. Lea todo el parcial antes de comenzar a responder.

**TEORÍA**

A -Explique la diferencia entre git add . y git add archivo.

git add . es para todos los cambios en el directorio, mientras que git add archivo es para un cambio específico en un archivo particular.

B -Explique la diferencia entre git checkout -b rama y git checkout rama

git checkout -b rama es para crear y cambiar a una nueva rama, mientras que git checkout rama es solo para cambiar a una rama que ya existe.

C -Describa brevemente el propósito de los comandos push y pull en Git.

git push envía los cambios al remoto, mientras que git pull trae los cambios del remoto al local.

D -¿Que es un hilo?

Un hilo es una secuencia de código en ejecución dentro del contexto de un proceso.

E - ¿Cuáles son los motivos de que un hilo este en estado not running?

Los hilos pueden pasar al estado not running a través de varias vías.

El método suspend()

El método sleep ()

El método wait()

El método interrupt()

F-



**public class Main{**

**public static void main(String[] args){**

**HiloVerdaderoFalso hilo = new HiloVerdaderoFalso();**

**hilo.start();**

**//Detener el hilo después de 5 segundos**

**try{**

**Thread.sleep(5000);**

**} catch (InterruptedException e) {**

**e.printStackTrace();**

**}**

**hilo.detenerHilo();**

**//Volver a arrancar el hilo**

**hilo = new HiloVerdaderoFalso();**

**hilo.start();**

**}**

**Afirmaciones:**

Afirmación 1: El hilo se detiene después de 5 segundos de ejecución.

Verdadera. El hilo se inicia con hilo.start() y luego se detiene después de 5 segundos debido a Thread.sleep(5000), seguido por la llamada a hilo.detenerHilo(). Suponiendo que el método detenerHilo() está implementado correctamente en la clase HiloVerdaderoFalso para detener el hilo, esta afirmación es verdadera.

Afirmación 2: Después de detener el hilo, se crea un nuevo hilo y se reinicia la ejecución.

La validez de esta afirmación depende de la implementación del método detenerHilo(). Si el método detenerHilo() simplemente cambia una bandera que el hilo verifica para detener su ejecución de manera segura, entonces la afirmación sería falsa, ya que es posible crear y empezar un nuevo hilo después de detener el anterior. Sin embargo, si se utilizara Thread.stop(), que está obsoleto y no se recomienda, entonces sería verdadero que el hilo no puede ser reiniciado, pero en el código proporcionado, se está creando una nueva instancia de HiloVerdaderoFalso y no se está reiniciando el hilo detenido.

Indica si cada afirmación es verdadera o falsa según el comportamiento observado del programa.

Justifica tus respuestas con base en la ejecución del código.

**PRÁCTICA**

1 -Inicie un nuevo repositorio Git llamado "mi\_proyecto".

git init

2 -Clone el repositorio "mi\_proyecto"

git clone https://github.com/Martincho83/mi\_proyectoRep.git

3 -Cree un archivo llamado "archivo.txt" y agregue algunas líneas de texto.

touch archivo.txt

nano archivo.txt

4 -Agregue "archivo.txt" al área de preparación (staging area).

git add archivo.txt

5 -Realice un commit con el mensaje "Agregando archivo.txt".

git commit -m "Agregando archivo.txt"

6 -Suba los cambios al repositorio

git branch -M main

git remote add origin https://github.com/Martincho83/mi\_proyectoRep.git

git push -u origin main

7 -Cree y cambie a una nueva rama llamada "nueva-caracteristica".

git checkout -b nueva\_caracteristica

8 -Modifique "archivo.txt" agregando más texto.

nano archivo.txt

9 -Agregue los cambios al área de preparación y realice un commit

con el mensaje "Agregando nueva característica".

git add archivo.txt

git commit -m "Agregando nueva caracteristica"

10 -Suba los cambios al repositorio

git push origin nueva\_caracteristica

11 -Cambie de regreso a la rama principal (master/main).

git checkout main

12 -Fusione la rama "nueva-caracteristica" con la rama principal.

git merge nueva\_caracteristica

13 -Resuelva cualquier conflicto que pueda surgir durante la fusión

15-El objetivo de esta tarea es implementar un programa en Java que utilice hilos para leer un archivo de texto de manera concurrente.

Implementación del Programa Java:

Deben crear un programa Java que utilice hilos para leer el archivo de texto proporcionado.

El programa debe crear al menos un hilo para llevar a cabo la lectura del archivo.

Mientras el hilo o los hilos leen el archivo, el hilo principal debe realizar otras operaciones, como mostrar un mensaje en la consola o realizar cálculos simples.

Firma……………………………………………………….