



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

Práctica 1

ALUMNO:

Martin Felipe Espinal Cruces - 316155362

PROFESOR

Salvador González Arellano

AYUDANTES

Rogelio Alcantar Arenas

Eric Toporek Coca

Luis Angel Leyva Castillo

ASIGNATURA

Computación Concurrente

25 de agosto de 2022

Pregunta 1

¿Por qué se pone un InterruptedException en el método main?

Cuando un hilo se encuentra dormido, en espera o en ejecución puede ocurrir que dicho hilo sea interrumpido antes o después de alguna actividad por lo que utilizar la excepción InterruptedException lo que nos permitirá cachar el error para evitar terminar el programa abruptamente ¹

Pregunta 2

¿Para que sirve el método Join?

Permite que un subproceso espere hasta que otro subproceso complete su ejecución. A pesar de que es el programador el que decide en que orden poner a esperar un hilo, es decisión del sistema operativo en que momento lo hace por lo que no debe suponer que la esperará exactamente el tiempo que especifique ²

Pregunta 3

¿Qué pasa si no le hacemos Join a los hilos?

Pondrá a los subprocesos actuales en espera hasta que el o los subproceso en el que se llama esté muerto. Si el subproceso se interrumpe, arrojará InterruptedException. ²

Pregunta 4

Explica de manera consisa como usar Hilos extendiendo la clase Thread.

Creas tu clase extendiendo a la clase Thread. Después tienes que implementar el método run() que será el que ejecutó el hilo. Posteriormente en el método main creas una instancia de tu clase y finalmente a esa instancia llamas al método start() que ejecutará el código que se encuentre en el método run() ³

```
// Ejemplo
public class Main extends Thread {
    public static void main(String[] args) {
        Main thread = new Main();
        thread.start();
        System.out.println("This code is outside of the thread");
    }
    public void run() {
        System.out.println("This code is running in a thread");
    }
}
```

Pregunta 5

¿Cuáles son las ventajas en implementar Runnable contra extender de Thread?

La principal ventaja de implementar la interfaz Runnable contra extender la clase Thread es que de esa forma podemos extender de otra clase lo que nos permite mantener la herencia. Cuando extendemos la clase Thread, cada uno de nuestros hilos crea un objeto único y se asocia con él. Cuando implementamos Runnable, comparte el mismo objeto con varios subprocesos. ⁴

Pregunta 6

¿Se puede predecir el orden en el que se imprimira el mensaje de la clase Hilos?

No ya que al existir múltiples hilos es decisión del Sistema Operativo o del JVM el orden y tiempo de ejecución de cada uno ⁴

Pregunta 7

En el archivo Hilos2.java, ¿Qué pasa si sacamos la instancia de la clase "h" de t_1 , es decir, poner h por ejemplo, antes de declarar t_1 ?

Nada, la ejecución continúa sin problema ya que la instancia de la clase "h" se utiliza en el método run()

Pregunta 8

Explica cómo podríamos tener comportamientos diferentes implementando Runnable

Si implementáramos la interfaz Runnable tendríamos el método run forzosamente afuera del método main y la instancia de la clase h tendría que estar declarada antes de la instancia de la clase Thread pero en cuestión de comportamiento serían similares

Bibliografía

1. InterruptedException (Java Platform SE 7). (s. f.). Docs Oracle. Recuperado 25 de agosto de 2022, de <https://docs.oracle.com/javase/7/docs/api/java/lang/InterruptedException.html>
2. GeeksforGeeks. (2021, 17 febrero). Joining Threads in Java. Recuperado 25 de agosto de 2022, de <https://www.geeksforgeeks.org/joining-threads-in-java/>
3. Java Threads. (s. f.). W3schools. Recuperado 25 de agosto de 2022, de https://www.w3schools.com/java/java_threads.asp
4. GeeksforGeeks. (2017, 4 octubre). Implement Runnable vs Extend Thread in Java. Recuperado 25 de agosto de 2022, de <https://www.geeksforgeeks.org/implement-runnable-vs-extend-thread-in-java/>