

PRACTICA 2

Hilos en Java

Como ejecutar

Para arrancar el codigo hay que primero crear el `.class` de cada clase, y despues ejecutar `java clase` para cada clase.

- `javac Herencia.java && java Herencia`
- `javac ClaseRunnable.java && java ClaseRunnable`

Preguntas

1. El código ejecuta 3 hilos, el padre, `h1` y `h2` que realizan una impresión de una palabra dada 10 veces, al ejecutar el programa el orden de ejecución de los hilos es aleatorio.
2. Al ejecutar el programa obtenemos el mismo resultado que usando herencia.
3. Al llamar al método `start` de cada hilo, estos comenzarán ejecutar el método `run`. Debido a que en java no hay herencia multiple, no podemos heredar de otras clases. Este problema desaparece si utilizamos la interfaces como `Runnable` para crear el hilo, ya que una clase puede implementar varias interfaces.
4. Para ajustar la prioridad de un hilo usaremos el metodo `setPriority`, de esta manera le damos mas prioridad al `h1` que al `h2`, por lo que en teoría el hilo2 se ejecutaría antes. Al probarlo en otro ordenador se ha obtenido otro orden, esto es debido a que la JVM no procesa bien las prioridades en nuestro sistema.

Hilos en Python

Como ejecutar

Para arrancar el codigo solo hay que pasarle al interprete de Python el archivo.

- `python3 hilos.py`

Preguntas

El programa parece funcionar correctamente, pero al ejecutarlo varias veces podemos observar discrepancia en los resultados, esto se debe a que hay una condicion de carrera entre los hilos al acceder al recurso, provocando un problema de sincronizacion.

Hilos en Rust

Como ejecutar

Para arrancar el codigo primero compilamos con `cargo build` y despues ejecutamos con `cargo run`. Aunque `cargo run` compila y ejecuta el programa.

- `cargo run`

Preguntas

En rust hemos usando Mutex y Arc, el programa parece funcionar correctamente, esto se debe a que Rust prevee condiciones de carrera, y permite la comparticion de recursos entre hilos de manera comoda.