

# Guía para la realización del ejercicio 4 del tema 4

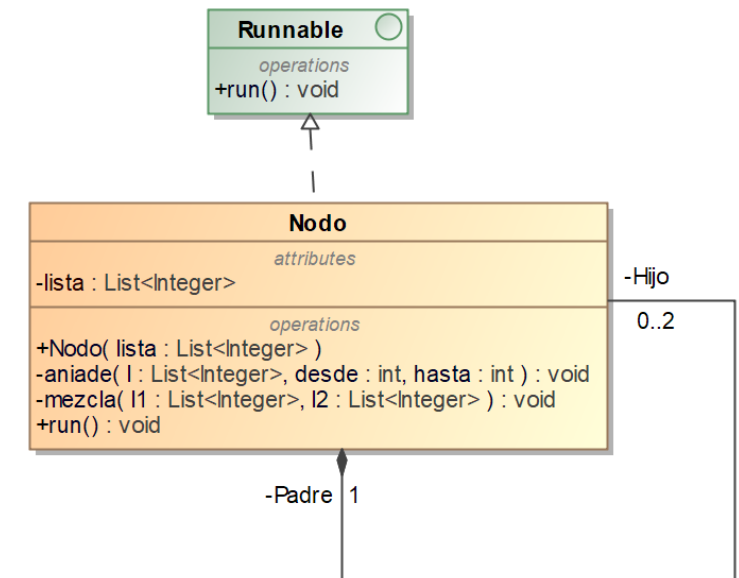
Programación de Sistemas y Concurrency

# Recomendaciones Generales

- Recuerda los ejercicios del bloque de concurrencia se realizan en Java
- Descarga la relación de ejercicios del tema 4 de la sala común
- Para cada uno de los ejercicios crea un proyecto Java diferente
- Para los ejercicios de esta relación no se proporcionan esqueletos, así que lee cuidadosamente cada uno de los enunciados e implementa todas las clases necesarias

# Ejercicio 4

- Diseña un programa Java que implemente el algoritmo recursivo de ordenación por mezcla de forma concurrente.
- El programa debe construir un árbol binario de hebras, de profundidad variable, dependiente del número de elementos de la lista a ordenar.
- El sistema sólo necesita una clase `Nodo` que hereda de `Thread`
- Implementar el método `main` en esta clase



# Ejercicio 4

-lista:List<Integer>

Lista de números a ordenar. Utilizar la clase `ArrayList` y sus métodos para el manejo de listas

+Nodo(lista:List<Integer>)

Constructor de la clase `Nodo`, recibe como parámetro la lista de números a ordenar

- añade(l:List<Integer>, int desde, int hasta):void

Añade a la lista `l` los elementos de `lista` comprendidos entre `desde` y `hasta` (ambos incluidos)

- mezcla(l1:List<Integer>, l2:List<Integer>):void

Mezcla de forma ordenada los elementos de `l1` y `l2` y deja el resultado en `lista`, donde `l1` y `l2` son listas ordenadas.

# Ejercicio 4

- + `run()` Implementa el comportamiento de la hebra
- Si `lista` tiene más de un elemento
  - Creamos dos nuevos Nodos (`n1`, `n2`), cada uno recibe la mitad de `lista` (`l1`, `l2`)
  - Comenzamos la ejecución de las hebras asociadas a `n1` y `n2`
  - Esperamos a que las hebras asociadas a `n1` y `n2` terminen
  - Mezclamos de forma ordenada las listas `l1` y `l2` y guardamos el resultado en `lista`

# Ejercicio 4

¿Qué debe hacer el método main?

- Crear una lista de números aleatorios
  - usar un objeto de la clase `Random` y llamar al método `nextInt`
- Crear un objeto `Nodo` y pasarle la lista anterior
- Comenzar la ejecución de la hebra asociada al objeto `Nodo`
- Esperar a que la hebra asociada al objeto `Nodo` termine
- Mostrar la lista

# Ejercicio 4: ejemplo de ejecución

