

# Reto 1

**Estudiante:** Jose Javier Bailón Ortiz

**Asignatura:** Programación de servicios y procesos

**Convocatoria:** 14/11/2023

## Sumario

A – Tic,Tac.....	3
A.1 Planteamiento del problema.....	3
A.1 Solución.....	3
A.2 Notas sobre el código.....	3
A.3 Ejecutar el código.....	3
B – Tic, Tac por grupos.....	4
B.1 Planteamiento del problema.....	4
B.1 Solución.....	4
B.2 Notas sobre el código.....	4
B.3 Ejecutar el código.....	5
C – Tic, Tac, Toc.....	6
C.1 Planteamiento del problema.....	6
C.1 Solución.....	6
C.2 Notas sobre el código.....	6
C.3 Ejecutar el código.....	7

# A – Tic,Tac

## A.1 Planteamiento del problema

El objetivo es tener un programa que imprima en pantalla de manera indefinida y en este orden:

TIC

TAC

...

Debe hacerse habiendo dos hebras, una para TIC, y otra para TAC.

## A.1 Solución

La solución establecer un turno estricto entre hebras. Para hacerlo ambas hebras comparten una variable común de la clase tictac.Reloj que establece a qué hebra le toca. Cada hebra comprueba si es su turno y si lo es imprime su palabra y cambia la variable compartida dándole paso a la otra hebra.

## A.2 Notas sobre el código

Las clases son:

- **MainTicTac:** Main del programa. Se encarga de crear las hebras(una para Tic y otra para Tac) y lanzarlas.
- **PasoDeReloj:** Extiende a Thread. Se encarga de imprimir la palabra. Es la clase de las hebras creadas en MainTicTac.
- **Reloj:** Contiene la variable que ambas hebras utilizan para sincronizar los turnos. También se encarga de limitar la cantidad de impresiones que se realizan finalizando una vez se llega al límite puesto por el usuario al inicio del programa

## A.3 Ejecutar el código

**Versión:** Java 21

**Paquete:** tictac

**Main:** tictac.MainTicTac

**Compilación:** javac tictac/MainTicTac.java

**Ejecución:** java tictac/MainTicTac

Tras la ejecución se preguntará el número de impresiones a realizar. Si se introduce un entero negativo se imprimirá indefinidamente hasta que el usuario cancele la ejecución (en consola ctrl+c )

## B – Tic, Tac por grupos

### B.1 Planteamiento del problema

El objetivo es tener un programa que imprima en pantalla de manera indefinida y en este orden:

TIC

TAC

...

Debe hacerse habiendo cuatro hebras agrupadas, en dos grupos. Dos hebras para TIC y dos para TAC.

Las hebras deben competir entre sí no pudiendo establecerse turnos estrictos entre hebras.

### B.1 Solución

La solución se basa en dos partes: por un lado asegurar la secuencia TIC,TAC,TIC,TAC... y por otro lado que no haya varios TAC o TIC consecutivos.

La solución está implementada con dos grupos de hebras y en cada grupo hay dos hebras que imprimen la misma palabra. Se va alternando el turno entre los grupos y en cada turno solo una de las hebras del grupo imprime en pantalla. Tras imprimir se pasa el turno al siguiente grupo.

Para asegurar la secuencia de grupos TIC, TAC.. he usado la solución del punto A para turnos estrictos pero en vez de alternar entre hebras alterna entre los grupos. Cada hebra mira si el turno corresponde al grupo al que pertenece. En caso de no corresponder hace una espera ocupada.

Para conseguir que solo una de las hebras imprima he usado una modificación del algoritmo de Dekker del cual he eliminado el turno de hebra para evitar que se convierta en una secuencia de turnos estrictos entre hebras del mismo grupo. Las hebras del mismo grupo comparten array de banderas con el cual coordinan la exclusión mutua de Dekker.

Ambos controles, el de turno de grupo y el de exclusión entre hebras del mismo grupo están mezclados en la condición que da paso a la impresión. Así las hebras hacen espera ocupada hasta que llegue el turno de su grupo y a la vez Dekker le da paso a una de las hebras.

### B.2 Notas sobre el código

Las clases son:

- **MainTicTacGrupos:** Main del programa. Se encarga de crear las hebras(dos para Tic y dos para Tac) crear los grupos, asignar un array de banderas a cada grupo y que comparten las hebras del mismo grupo. Una vez configurado todo eso las lanza.
- **PasoDeReloj:** Extiende a Thread. Se encarga de imprimir la palabra. Es la clase de las hebras creadas en MainTicTacGrupos.

- **Reloj:** Contiene la variable que todas las hebras utilizan para sincronizar los turnos entre grupos. También se encarga de limitar la cantidad de impresiones que se realizan finalizando una vez se llega al límite puesto por el usuario al inicio del programa.

## B.3 Ejecutar el código

**Versión:** Java 21

**Paquete:** tictac\_grupos

**Main:** tictac\_grupos.MainTicTacGrupos

**Compilación:** javac tictac\_grupos/MainTicTacGrupos.java

**Ejecución:** java tictac\_grupos/MainTicTacGrupos

Tras la ejecución se preguntará el número de impresiones a realizar. Si se introduce un entero negativo se imprimirá indefinidamente hasta que el usuario cancele la ejecución (en consola ctrl+c )

## C – Tic, Tac, Toc

### C.1 Planteamiento del problema

El objetivo es tener un programa que imprima en pantalla de manera indefinida y en este orden:

TIC

TAC

TOC

...

Debe hacerse habiendo seis hebras agrupadas, en tres grupos. Dos hebras para TIC, dos para TAC y dos para TOC.

Las hebras deben competir entre sí no pudiendo establecerse turnos estrictos entre hebras.

### C.1 Solución

La solución de este problema la he abordado modificando la solución del apartado B. Simplemente ha sido necesario agregar un nuevo grupo de hebras, el grupo toc, y tenerlo en cuenta en la secuencia de cambio de grupo. Ver apartado B para detalle en profundidad de la solución.

### C.2 Notas sobre el código

Las clases son:

- **MainTicTacToc:** Main del programa. Se encarga de crear las hebras(dos para Tic, dos para Tac y dos para Toc) crear los grupos, asignar un array de banderas a cada grupo y que comparten las hebras del mismo grupo. Una vez configurado todo eso las lanza.
- **PasoDeReloj:** Extiende a Thread. Se encarga de imprimir la palabra. Es la clase de las hebras creadas en MainTicTacToc.
- **Reloj:** Contiene la variable que todas las hebras utilizan para sincronizar los turnos entre grupos. También se encarga de limitar la cantidad de impresiones que se realizan finalizando una vez se llega al límite puesto por el usuario al inicio del programa. En este caso ha sido modificada respecto a la version del caso B agregando un nuevo paso en la secuencia que corresponde al grupo toc.

## C.3 Ejecutar el código

**Versión:** Java 21

**Paquete:** tictactoc

**Main:** tictactoc.MainTicTacToc

**Compilación:** javac tictactoc/MainTicTacToc.java

**Ejecución:** java tictactoc/MainTicTacToc

Tras la ejecución se preguntará el número de impresiones a realizar. Si se introduce un entero negativo se imprimirá indefinidamente hasta que el usuario cancele la ejecución (en consola ctrl+c ).