



## Programación de Sistemas y Concurrencia

## Control Bloque 2-Monitores

Informática A + Software A

Curso 2023-2024

APELLIDOS \_\_\_\_\_ NOMBRE \_\_\_\_\_

DNI \_\_\_\_\_ ORDENADOR \_\_\_\_\_ GRUPO/TITULACIÓN \_\_\_\_\_

### Descripción del sistema

Un grupo de **J** jugadores quieren participar en un torneo de padel. El torneo se juega en un club muy pequeño y **solo tienen disponible una pista de padel**. Por tanto, en cada momento solo 4 jugadores podrán jugar un partido y el resto tendrán que esperarse. **Los jugadores no tienen una pareja predefinida**, sino que cuando llegan al torneo se van agrupando de cuatro en cuatro para poder jugar un partido.

Cuando un jugador llega a la pista de padel, si hay hueco para el siguiente partido entrará en la pista de padel y se esperará hasta que sean cuatro. El último jugador en entrar avisará al resto de que ya se puede jugar el partido. Una vez que el partido termine los jugadores irán saliendo de la pista de padel y se esperarán hasta que salga el último. El último jugador en salir de la pista de padel informará de qué pareja ha sido la ganadora y avisará al resto de jugadores de que ya pueden seguir jugando si lo desean. A los dos jugadores de la pareja ganadora se les incrementará su puntuación en 10 puntos.

Después de cada partido, cada jugador decidirá si quiere continuar en el torneo o no. Si decide no seguir simplemente abandona el torneo. Si decide seguir intentará apuntarse a otro partido. El torneo finalizará cuando el número de jugadores que queden en el torneo sea menor de 4. Al acabar cada partido, el último jugador en salir de la pista comprobará el número de jugadores que quedan en el torneo. Si el número es mayor de 4 no hará nada, pero si es menor de 4 avisará de que el torneo ha finalizado e indicará quién es el ganador (el que haya acumulado mas puntos), si es que lo hay, sino informará de que ha habido un empate. Además, avisará al resto de jugadores que podrían estar esperando en la pista de padel para jugar un partido de que el torneo ha finalizado y deben salir.

Se pide implementar el juego descrito **utilizando métodos sincronizados o Locks**. En el campus virtual se proporcionan las siguientes clases:

- **Jugador**: extiende de Thread y recibe en el constructor el identificador del jugador y el torneo en el que participará. El código del método `run()` de esta clase indica que mientras no finalice el torneo, el jugador intenta formar un equipo de cuatro jugadores para jugar un partido (llamando al método `entrarEnPista`). Este método devuelve `false` si no ha sido posible formar un equipo porque en el torneo quedan menos de 4 jugadores, y devolverá `true` si el partido puede jugarse. Si se ha formado un equipo, se juega el partido y a continuación el jugador indica que ha finalizado (llamando al método `finPartido`). El jugador sigue jugando o no según el valor devuelto por el método `finPartido`. Si este método devuelve `true` el jugador no seguirá jugando. Si devuelve `false` el jugador seguirá jugando.
- **Driver**: esta clase es la encargada de crear el torneo y los jugadores. Crea **J** jugadores y un torneo.

Se pide implementar el constructor y los siguientes métodos de la clase **Torneo**, cuyo comportamiento debe ser el siguiente:

- **public Torneo(int totalJug)**: es el constructor de la clase. Recibe el número inicial de jugadores apuntados al torneo. En el constructor se inicializará el torneo. Tened en cuenta que habrá que almacenar los puntos de cada jugador, y que **cada jugador tendrá inicialmente 0 puntos**. Tened en cuenta también que para cada partido que se juegue, **se deben registrar los id de los cuatro jugadores en el partido**.
- **public boolean entrarEnPista(int id)**: este método es utilizado por el jugador `id` que quiere entrar a la pista de padel para jugar un partido. Tiene que esperar si ya hay un partido en juego (`CS1_Jugador`). Cuando haya hueco en el siguiente partido **registrará su id** y si no es el cuarto jugador en llegar tendrá que esperar hasta poder continuar (`CS2_Jugador`). Si es el cuarto en llegar

avisará al resto de que el partido puede continuar. Cuando avisan al jugador de que puede continuar han podido ocurrir dos cosas: (1) que le haya avisado el cuarto jugador en llegar al partido. Los dos primeros jugadores en llegar formarán la primera pareja y los otros dos la segunda pareja. En este caso este método devuelve `true` para indicar que se ha podido formar un equipo; o (2) que le haya avisado otro jugador que al terminar su partido se da cuenta de que quedan menos de 4 jugadores en el torneo y por tanto el torneo debe finalizar (ver descripción del método `finPartido()`). En este caso, este método devolverá `false` para indicar que el torneo ha finalizado y que no se ha podido formar un nuevo equipo de cuatro jugadores.

- **public boolean finPartido(int id):** este método es utilizado por el jugador `id` para informar que acaba de terminar de jugar un partido. Si no es el último jugador en abandonar la pista de padel se tendrá que esperar (`CS3_Jugador`). El último jugador le avisará cuando puede continuar. Una vez que le avise, tendrá que decidir si quiere seguir o no jugando. Para ello puede hacer uso del método privado `proximoJuego()` que devuelve `true` si sigue jugando y `false` en caso contrario. El último jugador en salir del partido será el encargado de: (1) averiguar e informar de quién ha sido la pareja ganadora del partido. También incrementará los puntos de los dos jugadores ganadores en 10. Se podrá hacer uso de los métodos privados `parejaGanadora()` (que devuelve 0 si ha ganado la primera pareja y 1 si ha ganado la segunda pareja) y `mostrarPuntos()` (que muestra los puntos de todos los jugadores que participan en el torneo); (2) decidir si él quiere continuar o no en el torneo, y (3) comprobar si se ha llegado o no al final del torneo (número de jugadores que quedan en el torneo es menor de 4). Si se ha llegado al final del torneo el último jugador averiguará quién ha sido el ganador del torneo usando el método privado `obtenerGanador()` (que devuelve el `id` del ganador o -1 si ha habido un empate en la máxima puntuación y no hay ganador). Además, informará al resto de jugadores que el torneo ha terminado y él también saldrá, independientemente de la decisión que hubiera tomado anteriormente de continuar o no. A los jugadores que podrían estar esperando para jugar un partido les avisará de que el torneo ha finalizado y por tanto deben continuar, pero para salir del torneo.

Las condiciones de sincronización que hay que implementar son las siguientes:

- (1) `CS1_Jugador`: Un jugador que intenta entrar a la pista de padel para jugar un partido espera si hay otro partido en juego.
- (2) `CS2_Jugador`: Un jugador que ya ha conseguido apuntarse a un partido espera si aún no hay cuatro jugadores en el partido.
- (3) `CS3_Jugador`: Un jugador que termina de jugar un partido y sale de la pista de padel espera hasta que el último jugador del partido abandone la pista de padel.

Una posible salida de la ejecución del juego con 10 jugadores podría ser:

Jugador 0 llega al partido	Jugador 1 llega al partido
Jugador 9 llega al partido	Jugador 5 llega al partido
Jugador 8 llega al partido	Jugador 4 llega al partido
Jugador 7 llega al partido	JUGADORES [2, 1, 5, 4]
JUGADORES [0, 9, 8, 7]	Han ganado los jugadores 5 y 4
Han ganado los jugadores 0 y 9	PUNTOS: [10, 0, 10, 0, 10, 10, 0, 0, 0, 20]
PUNTOS: [10, 0, 0, 0, 0, 0, 0, 0, 10]	Quedan 3 jugadores en el torneo.
Quedan 9 jugadores en el torneo.	*****
*****	No hay jugadores suficientes. Torneo finalizado!
Jugador 6 llega al partido	*****
Jugador 0 llega al partido	PUNTOS: [10, 0, 10, 0, 10, 10, 0, 0, 0, 20]
Jugador 9 llega al partido	El jugador 9 es el ganador del torneo
Jugador 2 llega al partido	Jugador 4 avisa a los que estén esperando y se va
JUGADORES [6, 0, 9, 2]	Jugador 3 llega al partido
Han ganado los jugadores 9 y 2	Jugador 3 se va, el torneo ha finalizado
PUNTOS: [10, 0, 10, 0, 0, 0, 0, 0, 20]	Jugador 5 llega al partido
Quedan 5 jugadores en el torneo.	Jugador 5 se va, el torneo ha finalizado
*****	Jugador 1 llega al partido
Jugador 2 llega al partido	Jugador 1 se va, el torneo ha finalizado

## FIN DEL ENUNCIADO