

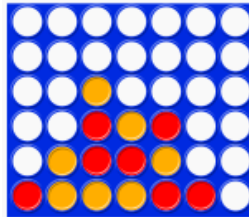


## Ejercicio Obligatorio. Sesión 01.

Debe estar implementado antes de la siguiente clase de laboratorio.

### Enunciado

El juego denominado "Conecta-4" (también conocido como "4 en raya") es un juego de mesa para dos jugadores, en el que se introducen fichas en un tablero vertical con el objeto de alinear cuatro fichas consecutivas de un mismo color. Se gana al alinear 4 fichas del mismo color en una fila, una columna o en diagonal.



En el juego de mesa, el tablero consta de 6 filas y 7 columnas y cada jugador dispone de 21 fichas. Para implementar este juego se utilizará un array bidimensional de caracteres. A continuación, se muestra el aspecto que podría tener una partida:

```
0 1 2 3 4 5 6
. . . . . . .
. . * + . . .
. . + + + * .
. . * * + + .
. . * + * + .
. + * * * + *
```

Para el ejercicio a implementar podemos suponer que el punto es para indicar un hueco, el primer jugador empleará el asterisco y el segundo el signo de suma. Lo único que se tendrá que indicar es la columna en la que desean "dejar caer" la ficha. Por ejemplo,

```
0 1 2 3 4 5 6
. . . . . . .
. . * + . . .
. . + + + * .
. . * * + + .
. . * + * + .
. + * * * + *
```

Turno del jugador '\*'

¿En qué columna desea introducir la ficha? 6

```
1 2 3 4 5 6
. . . . . . .
. . * + . . .
. . + + + * .
. . * * + + .
. . * + * + *
. + * * * + *
```

Obsérvese que la ficha recién introducida cae hasta "chocar" con una ficha anterior o con la parte inferior del tablero. El juego finaliza cuándo hay un ganador, indicando quién es el ganador o cuándo no hay ganador, pero el juego ha finalizado porque ya no se pueden colocar más fichas.

Se debe completar el proyecto proporcionado a partir de las siguientes especificaciones:

1. Clase **Conecta4** que incluya las constantes necesarias y las propiedades **winnerPlayer**, **currentPlayer** y **board** como matriz de caracteres con los siguientes métodos públicos:



2. Un constructor sin parámetros ***public Conecta4()*** (para crear un tablero de dimensión fija 6x7) y otro **constructor con dos parámetros** ***public Conecta4(int higt, int width)*** para indicar las filas y columnas del tablero (y así poder tener diferentes dimensiones).
3. Método ***public char getCurrentPlayer()*** devuelve un carácter (asterisco o suma) indicando el turno del jugador.
4. Método ***public boolean play(int column)*** que simula el turno en el que un jugador introduce una ficha en una columna del tablero. Este método devuelve **true** si se pudo colocar la ficha y **false** en caso contrario. Si se puede colocar la ficha se cambia además el turno del jugador.
5. Método ***public boolean finished()*** que indica si se ha llegado o no al final de la partida. Es el final de la partida cuando hay un ganador o ya no hay más huecos en el tablero para colocar fichas. Este método devuelve **true** si el tablero está completo (no hay huecos) o hay un jugador ganador, **false** en caso contrario. Habrá un jugador ganador si al examinar las filas, columnas o diagonales se encuentran cuatro fichas consecutivas del mismo color (en nuestro caso cuatro '\*' o '+').
6. Método ***public char winner()*** que determina quién es el jugador ganador o si quedaron en tablas. Devuelve un carácter representando la ficha del ganador o el carácter hueco indicando que hubo tablas.
7. Método ***public String toString()*** que devuelve una cadena de texto con la información del tablero por líneas.
8. **Generar la documentación con JavaDoc.** El código del proyecto debe incluir comentarios de documentación y comentarios de implementación.