

Actividad Entregable UT3

No utilices un jdk superior al 17. Recuerda que debes entregar el proyecto que pueda abrirse en NetBeans correctamente. No es necesario la generación de ningún .jar.

Actividad 1 (6,5 puntos)

Implementa un **servidor TCP y un cliente TCP que van a jugar a las 3 en raya**. Un cliente TCP se conecta al servidor TCP para jugar teniendo en cuenta que:

- El servidor TCP siempre está ejecutándose esperando a que un cliente nuevo se conecte.
- El servidor TCP estará alojado en localhost y escuchando en un puerto adecuado. El cliente TCP conoce la dirección y el puerto del servidor al que va a conectarse.
- Cuando un cliente TCP se conecta al servidor TCP le envía un mensaje de bienvenida.
- El cliente TCP imprimirá por pantalla todos los mensajes que le lleguen del servidor TCP.
- El cliente TCP es el que empieza la partida. Solicita dos coordenadas al usuario, la fila y la columna y no realices ninguna comprobación sobre ellas. Será el servidor TCP el encargado de comprobarlo, pero el cliente TCP no debe fallar porque se introduzca algo incorrecto. La fila y columna introducidas son enviadas al servidor TCP.
- El servidor TCP debe comprobar que las filas y columnas tienen una numeración correcta, es decir, como el tablero es de 3x3, deben ser números del 0 al 2. En caso de que no sean números o no sea una numeración correcta el servidor termina la partida enviando un mensaje al cliente TCP. El cliente TCP finalizaría.
- El servidor TCP también finaliza el juego si la fila y columna proporcionada por el cliente TCP ya se encuentra ocupada.
- Si la jugada del cliente TCP es ganadora, el servidor TCP informa al usuario y termina la partida.
- Si la jugada del cliente TCP no es ganadora, el servidor TCP realiza una jugada aleatoria.
- Si la jugada del servidor TCP es ganadora, el servidor TCP informa al usuario y termina la partida.
- Si la jugada del servidor TCP no es ganadora, se informa al cliente TCP y quedamos a la espera de la siguiente jugada.
- Es posible que el juego termine en empate, es decir, no se pueden seguir rellenando el tablero, el servidor TCP informa al usuario y termina la partida.

- Siempre se envía el tablero actual al cliente TCP para que este pueda comprobar como va la partida.
- El servidor TCP debe imprimir mensajes en su consola para comprobar que está haciendo.
- Cuando el servidor TCP termina la partida con un cliente TCP queda a la espera de que se conecte otro cliente TCP.
- Observa (no tienes que hacer nada) que si varios clientes deciden jugar a la vez, el servidor los va atendiendo de uno en uno. Solucionaremos esto en la unidad 4.

A continuación, se adjunta diferentes simulaciones del juego. Puedes utilizarlas como referencia para tu manual.

Cliente gana la partida

Cliente	Servidor
<pre> Servidor del juego: ¡¡¡Bienvenido al juego!!! Envía un número de fila o columna Introduce la fila: Introduce columna Servidor de juegos: Seguimos jugando - - - - X - - O - Introduce la fila: Introduce columna Servidor de juegos: Seguimos jugando X - - O X - - O - Introduce la fila: Introduce columna Servidor de juegos: ¡Has ganado! X - - O X - - O X Process finished with exit code 0 </pre>	<pre> Soy el servidor TCP esperando a que alguien se conecte ¡Se ha conectado un nuevo jugador! Fila recibida 1 Columna recibida 1 Después de la jugada del cliente - - - - X - - - - Después de la jugada del servidor - - - - X - - O - Fila recibida 0 Columna recibida 0 Después de la jugada del cliente X - - - X - - O - Después de la jugada del servidor X - - O X - - O - Fila recibida 2 Columna recibida 2 Después de la jugada del cliente X - - O X - - O X El cliente ha ganado Soy el servidor TCP esperando a que alguien se conecte </pre>

Cliente pierde la partida y gana el servidor

Cliente	Servidor
<pre>Servidor del juego: :::Bienvenido al juego!!! Envía un número de fila o columna Introduce la fila: Introduce columna Servidor de juegos: Seguimos jugando X - - - - - - o - Introduce la fila: Introduce columna Servidor de juegos: Seguimos jugando X X - - - - - o o Introduce la fila: Introduce columna Servidor de juegos: ¡SKYNET ha ganado! X X - - X - o o o Process finished with exit code 0</pre>	<pre>Soy el servidor TCP esperando a que alguien se conecte ¡Se ha conectado un nuevo jugador! Fila recibida 0 Columna recibida 0 Después de la jugada del cliente X - - - - - - - - Después de la jugada del servidor X - - - - - - o - Fila recibida 0 Columna recibida 1 Después de la jugada del cliente X X - - - - - o - Después de la jugada del servidor X X - - - - - o o Fila recibida 1 Columna recibida 1 Después de la jugada del cliente X X - - X - - o o Después de la jugada del servidor X X - - X - o o o El servidor ha ganado Soy el servidor TCP esperando a que alguien se conecte</pre>

Cliente y el servidor empatan

Nota: He puesto sólo el final, que es cuando se produce el empate.

Cliente	Servidor
---------	----------

<pre> Introduce la fila: [Introduce columna] Servidor de juegos: Seguimos jugando X o X X o - o X o Introduce la fila: [Introduce columna] Servidor de juegos: Hay un empate de SKYNET y el cliente X o X X o X o X o Process finished with exit code 0 </pre>	<pre> Fila recibida 1 Columna recibida 0 Después de la jugada del cliente X o X X o - o X - Después de la jugada del servidor X o X X o - o X o Fila recibida 1 Columna recibida 2 Después de la jugada del cliente X o X X o X o X o Hay un empate de SKYNET y el cliente Soy el servidor TCP esperando a que alguien se conecte </pre>
---	--

Cliente envía filas y columnas incorrectas

Cliente	Servidor
<pre> Servidor del juego: ¡¡¡Bienvenido al juego!!! Envía un número de fila o columna Introduce la fila: [Introduce columna] Servidor de juegos: Aprende a jugar - - - - - - - - - Process finished with exit code 0 </pre>	<pre> Soy el servidor TCP esperando a que alguien se conecte ¡Se ha conectado un nuevo jugador! Fila recibida 1 Columna recibida asas El cliente ha introducido valores erróneos. Terminará el juego. Soy el servidor TCP esperando a que alguien se conecte </pre>

Cliente envía una fila y columna ya ocupada

Cliente	Servidor
<pre> Servidor del juego: ¡¡¡Bienvenido al juego!!! Envía un número de fila o columna Introduce la fila: [Introduce columna] Servidor de juegos: Seguimos jugando - o - - X - - - - Introduce la fila: [Introduce columna] Servidor de juegos: Aprende a jugar - o - - X - - - - Process finished with exit code 0 </pre>	<pre> Soy el servidor TCP esperando a que alguien se conecte ¡Se ha conectado un nuevo jugador! Fila recibida 1 Columna recibida 1 Después de la jugada del cliente - - - - X - - - - Después de la jugada del servidor - o - - X - - - - Fila recibida 0 Columna recibida 1 El cliente ha introducido valores erróneos. Terminará el juego. Soy el servidor TCP esperando a que alguien se conecte </pre>

Nota: Adjunto una clase con la lógica del tablero por si te sirve de ayuda para resolver el ejercicio. No tienes por qué usarla y puedes programarlo a tu manera. Tablero.java.

Actividad 2 (3,5 puntos)

Vamos a crear un **servidor UDP y cliente UDP**. El servidor UDP genera palabras aleatorias de una longitud dada por el cliente que lo solicita:

- El servidor UDP está continuamente esperando peticiones de palabras aleatorias, es decir, está continuamente ejecutándose.
- El cliente conoce la dirección y el puerto del servidor UDP. Utiliza localhost cuando subas la tarea. Elige un número de puerto adecuado.
- El cliente UDP solicita una longitud (un número entero) al usuario y se la envía al servidor. El cliente UDP no debe hacer ningún tipo de comprobación del número introducido.
- El servidor UDP recibe un número con la longitud de la palabra aleatoria que debe generar. Si los bytes recibidos no corresponden con un valor numérico o es un número negativo se le informa al usuario.
- El servidor UDP devuelve al cliente UDP la palabra aleatoria generada.
- El cliente UDP recibe la palabra generada o la cadena de error y se imprime por pantalla.
- El cliente UDP termina cuando imprime la respuesta del servidor UDP.
- El servidor UDP debe imprimir mensajes por pantalla para que se muestre lo que está haciendo.

A continuación, se adjunta diferentes simulaciones del programa. Puedes utilizarlas como referencia para tu manual.

Cliente envía un 5

Cliente	Servidor
<pre>Escriba la longitud de la palabra: 5 Respuesta del servidor: dSKAq Process finished with exit code 0</pre>	<pre>El servidor UDP queda a la espera de peticiones Recibida petición de un cliente: 5 Es un número válido genero la palabra dSKAq El servidor UDP queda a la espera de peticiones</pre>

Cliente envía un -5

Cliente	Servidor
---------	----------

<pre> Escriba la longitud de la palabra: -5 Respuesta del servidor: No ha enviado una longitud válida Process finished with exit code 0 </pre>	<pre> El servidor UDP queda a la espera de peticiones Recibida petición de un cliente: -5 No ha enviado una longitud válida El servidor UDP queda a la espera de peticiones </pre>
--	--

Cliente envía asdads

Cliente	Servidor
<pre> Escriba la longitud de la palabra: asdads Respuesta del servidor: No ha enviado una longitud válida Process finished with exit code 0 </pre>	<pre> El servidor UDP queda a la espera de peticiones Recibida petición de un cliente: asda No ha enviado una longitud válida El servidor UDP queda a la espera de peticiones </pre>

Lo que debes entregar:

- Los proyectos NetBeans de las dos actividades bien comentados. No es necesario Javadoc ni generar el .jar. Cada ejercicio debe estar en una carpeta distinta, actividad-1 y actividad-2.
- Manual con las pruebas realizadas. Realiza un sólo manual con las dos actividades. Únicamente es necesario los pantallazos de la salida de tu programa y una pequeña explicación de lo que ocurre o si has utilizado algún argumento de entrada a tu programa que deba poner también. Deben verse pruebas representativas de lo que hacen los programas.