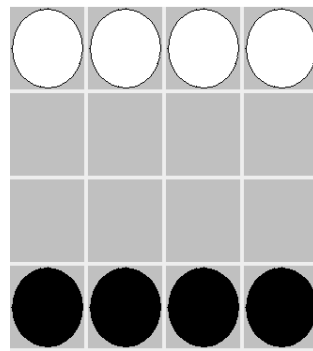


## Enunciado

**PARTE 1 (3 puntos)** Se quiere añadir al programa desarrollado durante el curso el juego *Peones* que se juega entre dos jugadores (a los que denominamos blancas y negras). Consiste en un tablero  $4 \times 4$  en el que inicialmente están situados 4 peones de cada color, ocupando cada color una fila, los peones blancos en la fila superior del tablero y los negros en la fila inferior, como se muestra en la siguiente figura:



Las reglas del juego son las siguientes:

- **Inicio del juego.** El jugador que inicia el juego debe ser elegido de forma aleatoria.
- **Movimientos.** En cada turno, el jugador puede mover un peón de su color siguiendo las reglas de movimiento del peón de ajedrez (simplificadas), avanzando a una casilla adyacente hacia el lado del adversario (las blancas hacia abajo; las negras hacia arriba), en vertical o diagonal siguiendo una de las siguientes reglas:
  - Si la siguiente casilla en vertical está vacía, se puede mover a esa casilla;
  - Si en diagonal (en el sentido de avance) hay un peón del color contrario, se puede mover a esa casilla, capturando el peón contrario (eliminándolo del tablero).
- **Siguiente jugador.** Los dos jugadores se van alternando siempre que se pueda. Si un jugador no puede realizar ningún movimiento, se pasa el turno al oponente.
- **Fin del juego.** Se pueden producir los siguientes finales de juego:
  - Un jugador **gana** si consigue situar un peón en la fila inicial del otro jugador (es decir, un peón negro consigue llegar a la fila superior del tablero, o un peón blanco consigue llegar a la fila inferior).
  - Se producen **tablas** si ninguno de los dos jugadores puede mover ningún peón.

Añade un paquete **peones** dentro de **practica4** con las clases necesarias para que se pueda jugar a este juego en modo consola, además de los juegos ya disponibles en la práctica 4. Modifica **examenJunio.Main** para poder iniciar el programa jugando a **peones** usando la opción “-g peones” desde la línea de comandos.

**PARTE 2 (1,5 puntos)** Añade un paquete `peones` en `practica5` con las clases necesarias para que se pueda jugar a este juego en el interfaz de ventana. Recuerda que el juego debe funcionar también en modo cliente-servidor.

**PARTE 3 (2,5 puntos)** Se quiere implementar un nuevo botón denominado `bomba` con la siguiente funcionalidad:

1. debe elegir una casilla del tablero de forma aleatoria que contenga una ficha del jugador (la denominamos *origen*);
2. debe aplicar la siguiente operación 10 veces: elegir dos posiciones aleatorias a una distancia máxima  $d$  del *origen* e intercambiar sus valores si ninguna de ellas es un obstáculo.

Modifica la vista de ventana de `practica5` añadiendo un combo-box con la potencia  $d$  de la bomba (valores de 1 a 5) y el botón `bomba`, que puede utilizar el jugador que tenga el turno en lugar de un movimiento normal. En modo multiventana o cliente-servidor este botón solo debe estar habilitado cuando el jugador tiene el turno. Recuerda que el botón `bomba` debe funcionar correctamente también en modo cliente-servidor.

**PARTE 4 (3 puntos)** En el servidor de juego de `practica6` todas las fichas se asignan a clientes. Se quiere modificar esta funcionalidad para permitir marcar algunas fichas como *random* en lugar de asignarlas a clientes. Después de iniciar el juego, cuando sea el turno de una ficha *random*, el servidor realizará automáticamente un movimiento utilizando un jugador aleatorio. De esta forma, es posible por ejemplo iniciar un juego con dos clientes y dos jugadores aleatorios.

Para proporcionar esta funcionalidad, añade un botón denominado “*Random Player*” a la ventana del servidor que cuando se pulse haga lo siguiente:

- La siguiente ficha pendiente de asignar será marcada como *random* y no será asignada a ningún cliente.
- Se mostrará un mensaje en el área de información de la ventana del servidor, por ejemplo: “*Piece X is marked as random.*”
- Si después de marcar la ficha como *random* todas las fichas han sido asignadas (a clientes o como *random*), el juego deberá iniciarse.

El botón “*Random Player*” debe estar habilitado solo cuando el juego no esté iniciado. Como se ha mencionado, cuando sea el turno de una ficha *random* el servidor deberá realizar automáticamente un movimiento utilizando un *random player*. Cuando termine el juego, todas las fichas volverán a estar no asignadas a clientes o *random*. No debes tener en cuenta el caso en el que un cliente se conecte exactamente al mismo tiempo que se pulsa el botón.

Además, después de enviar la notificación `onGameStart` a todos los clientes, se debe enviar un nuevo tipo de notificación denominado `randomPlayers(List<Piece> l)` a todos los clientes en el que se les pasa la lista de fichas *random*. Para implementar esta parte debes añadir primero un método nuevo `public void randomPlayer(List<Piece> l)` al interfaz `GameObserver`, y modificar todas las clases que lo implementan (tanto en `basecode` como en tus prácticas) para incluir ese método vacío. Después, modifica los métodos de las vistas de Swing para mostrar la lista de fichas *random* en el área “**Status Messages**” de la ventana. Modifica `practica6` para que el servidor pueda enviar estas modificaciones a los clientes.