

A continuación, se mostrará una secuencia de movimientos a seguir en la metodología del juego checkers, para así evidenciar cada una de sus funcionalidades y adicionalmente mostrar las características de cada ficha que la hace especial de las demás.

- Hacer visible al tablero con el método `makeVisible`.
- Agregar las fichas como una cadena, en el tablero con el método `read`:  
`("-.--.-Wb-.---Pw-.--Zb-.Nb---Nw-.--.-.-Hw-Nw-Nb.-.-LB-.")`  
 para tener mas claridad en las posiciones de la cadena se especifica a continuación la referencia que hace cada una.
  - Tipo: powerful, color: (negra), posición: (2,3)
  - Tipo: proletarian, color: (blanca), posición : (3,4)
  - Tipo: lazy, color: (negra), posición: (4,3)
  - Tipo: normal, color: (blanca), posición (5,4) : (7,6)
  - Tipo: hurried, color: (blanca), posición: (7,4)
  - Tipo: normal, color: (negra), posición: (4,7)
  - Tipo: normal, color: (negra), posición: (7,8)
  - Tipo: libertarian: (Rey), color : (negra), posición (8,5)
- Verificamos que no se puede agregar una ficha en una posición que ya se encuentra:  
`add(false,false,2,3)`
- Guardamos la posición inicial del tablero:  
`save("tableoInicial")`
- Realizamos el cambio de tablero de configuración al tablero de juego:  
`Swap()`

En este punto queremos dar a conocer el comportamiento de cada ficha, para tener mas claridad en este punto, primero realizamos una breve descripción de cada ficha y luego si realizamos los diferentes movimientos en el tablero de juego para ver su funcionalidad.

- Comportamientos de las fichas:
  - Rey: puede moverse en cualquier dirección.
  - Normal: realiza los movimientos básicos sin ninguna característica especial.
  - Proletaria: muere al tener que convertirse en rey.
  - Lazy: en un movimiento, solo salta una vez.
  - Libertarian: no captura las piezas que salta.
  - Powerful: no se deja capturar.
  - Hurried: repite dos veces el movimiento, si puede.

- Movimientos de las fichas:
  - En este punto realizamos el comportamiento de las fichas "proletarian" y "powerful", para ello seleccionamos la ficha "proletarian" :  
`select(3,4)`  
`jump(true,false)`  
 se puede evidenciar que al saltar la ficha "proletarian" muere al convertirse en rey y la ficha "powerful" no muere por que no se deja capturar por otra ficha.
  - En este punto realizamos el comportamiento de la ficha lazy, llamando al método move con diferentes saltos:  
`move("14x23x32")`  
 al realizar los saltos se evidencia que la ficha lazy solo puedo generar un salto y no logra matar a la otra ficha.
  - En este punto realizamos el comportamiento de la ficha hurried:  
`select(7,4)`  
`jump(true,true)`  
 se puede evidenciar que genera dos saltos, ya que puede repetir su movimiento.
  - En este punto realizamos el comportamiento de una ficha normal, que al moverse a la casilla inicial del oponente se convierte en rey:  
`select(7,8)`  
 se puede evidenciar que la ficha se convirtió en rey y por ende se puede mover en cualquier dirección.
  - En este punto realizamos el comportamiento de la ficha "libertarian":  
`select(8,5)`  
`jump(true,true)`  
 se puede evidenciar que la ficha no puede capturar a las fichas que salta.
  - Verificamos que la ficha que se convirtió en rey se puede mover en cualquier dirección:  
`select(8,7)`  
`shift(false,false)`  
 evidenciamos que se puede mover en cualquier dirección y que además pudo matar a la ficha opuesta.

#### Otros métodos

- en este punto la idea es guardar los estados del juego inicial y final, para eso realizamos el llamado a los siguientes métodos:  
`swap`  
`save("tableoFinal")`  
`recover("tableoInicial")`  
`swap`  
`recover("tableoFinal")`