

Universidad de Costa Rica

Escuela de Ciencias de la Computación e Informática

Diseño de Software

CI-0136

Primera iteración del MARDA

Profesor:

Alan Calderón Castro

Integrantes:

Rodrigo Li Qiu B94263

Axel Matus B75497

Jorim G. Wilson Ellis B98615

Asdrúbal Villegas Molina B88583

II Ciclo 2021

Clase **Checkers** es la clase que contiene el main. Su función es asignar los valores a la clase board e iniciar el juego, osea que crea la ventana gráfica, asigna el tablero a la ventana creada y fija los límites de dicha ventana en la pantalla.

Clase **Board** contiene lo pertinente al tablero como los movimientos a realizar o realizados, el fin e inicio de un juego, entre otras cosas y la parte gráfica en donde dichas eventualidades se plasman en la pantalla de los jugadores.

Clase **Data** contiene la información correspondiente al juego en proceso y las reglas del mismo. Esta clase se encarga de mover las piezas del juego y asegurarse de que los movimientos sean legales por cada jugador dentro del juego.

Clase **MovesMade** Esta clase es subordinada de la clase data y proporciona información sobre los movimientos que piensa hacer el jugador, donde analiza cuáles movimientos se pueden realizar, si tiene o no que comer y si su movimiento fue un salto.

Parte de este programa fue extraído de <https://github.com/tobyjamesthomas/Java-Checkers>, en donde le hicimos algunas modificaciones según nuestras necesidades. La primera decisión fue disminuir el número de clases de 5 a 4 ya que una de estas (la clase blah) no hacía nada relevante y alguna de las otras demás clases podían fácilmente cumplir su función. Otra decisión fue agregar la función de “game over”, o mejor dicho, indicarle al usuario de manera más clara que el juego ha terminado y debe iniciar uno nuevo, Cambiamos lo de pedir nombres para en cambio pedir color deseado del jugador (negro o blanco), además acomodamos el código para que se adaptara más a los patrones de diseño vistos en clase, documentamos todo el código con el estandar de javadocs y cambiamos la función de botón de información para que desplegara las reglas del juego. La última modificación es que se agrandaron las dimensiones del juego ya que lo considerábamos muy pequeño, sin embargo, cabe aclarar que el juego luce diferente según la resolución de la pantalla.

Algunas decisiones de diseño que se querían implementar en esta iteración, pero no se pudieron por falta de tiempo son:

- Modularizar más el código
- Pasar a privado algunos atributos de clases que son públicas (siempre es buena práctica mantener los atributos privados y que otras clases interactúen con estos mediante métodos como getters y setters).
- Intentar agregar más patrones de diseño