

## **IC-2101 Programación Orientada a Objetos**

***Prof. Mauricio Avilés***

### ***Tarea Programada III – Implementación de Maharajá y los cipayos***

#### ***Ajedrez***

#### ***¿Qué se busca con este proyecto?***

El objetivo de este proyecto es elaborar la implementación del segundo proyecto del curso Programación Orientada a Objetos, el cual consistió en la especificación del modelo UML de una aplicación para jugar.

#### ***Proyecto a desarrollar***

Con base en el modelo desarrollado en el proyecto anterior, debe construirse un programa Java que implemente las clases propuestas y brinde una interfaz de usuario que permita jugar una partida de Maharajá y los Cipayos entre dos personas. Se requiere que la interfaz de usuario sea gráfica y que tenga muy buena usabilidad y sea fácil de entender.

El juego debe mostrar al usuario el estado del tablero en cada jugada. En cada turno, el jugador indica al programa una celda origen y una celda destino que indica el movimiento que quiere realizar. El programa debe revisar la validez de dicha jugada, aplicando las reglas del juego. Si la jugada no es válida, debe darse un mensaje para notificar al usuario y debe solicitar la repetición de la jugada. Si la jugada es válida, el programa debe modificar el estado del juego para reflejarla, y debe imprimir el resultado obtenido después de la jugada: si sólo sucede el movimiento, si se capturó una pieza o si se produce un jaque.

El jugador debe tener la posibilidad de abandonar una partida antes de iniciar su turno.

El juego debe evitar que el usuario pueda mover al rey o maharajá a una posición de jaque. Si eso se intenta, debe considerarse como una jugada inválida. En dado caso, debe informarse al usuario e imprimir un mensaje en pantalla.

No es necesario que el programa indique cuándo sucede un jaque mate. Si el juego no permite movidas que causen jaques en el rey propio, entonces en caso de jaque mate no va a tener ninguna movida válida, por lo que su única opción será abandonar la partida.

Al finalizar una partida, debe mostrarse un mensaje con el motivo por el que terminó, sea abandono o tablas. El juego debe declararse tablas cuando los dos jugadores están de acuerdo.

## ***Documentación***

El código fuente debe tener documentación interna, con comentarios precisos y bien ubicados.

En cuanto a la documentación externa, debe entregarse dos documentos en formato PDF:

1. Manual de usuario: explica detalladamente a cualquier usuario cómo ejecutar y utilizar la aplicación. Debe tener portada.
2. Documentación del proyecto
  - a. Portada
  - b. Resumen ejecutivo. Descripción breve de todo lo que abarca la documentación. El objetivo de este resumen es captar la atención del lector y motivarlo a aprender más sobre el proyecto. Menos de una página.
  - c. Diagrama de clases con detalles de la implementación, usando notación UML. El diagrama debe estar incluido en el documento. Este diagrama debe ser diferente al entregado en el segundo proyecto, es inverosímil que no se produzcan cambios o ajustes al pasar del modelado a la implementación.
  - d. Pruebas de funcionalidad
    - i. Debe demostrar evidencia de todas las funcionalidades implementadas en la tarea programada. Para esto realice la ejecución de su tarea programada y tome *screenshots* dejando evidencia del resultado de la ejecución de cada transacción.
  - e. Conclusiones y recomendaciones. Deben ser de carácter técnico, sobre el proceso de traducir el modelo a una implementación concreta y sobre la programación en lenguaje Java. Evite referirse a la experiencia personal de los miembros del equipo.
  - f. Referencias. Fuentes consultadas para el desarrollo del proyecto. Deben incluirse en formato APA.

## ***Evaluación***

La evaluación se va a centrar en tres elementos: diagrama de clases, programación y documentación. El diagrama de clases detallado debe presentarse antes que finalice la primera semana del proyecto para ser aprobado por el personal del curso (profesor o asistentes). Esta revisión es necesaria y obligatoria, y debe darse antes del inicio de la programación. Si no se presenta, se pierde el 20% correspondiente.

Programación	80%
Documentación	20%

## ***Forma de trabajo***

El trabajo se debe realizar en parejas o tríos. No se permiten trabajos individuales.

## ***Entrega***

El tiempo asignado para la tarea programada es de 3 semanas.

El lenguaje de programación a utilizar es Java, y debe ser desarrollada en el IDE NetBeans 8.