

Rúbrica de Evaluación

Presentación (video) - 25%

Se expone el trabajo con claridad. Se hace excelente uso de recursos digitales, por ejemplo: gráficos o diagramas (de ser relevante). Se incluyen trozos de código que complementan la presentación (NO TODO EL CÓDIGO, solo fragmentos relevantes). No se excede en texto escrito. Se ajusta al tiempo asignado (10–15 minutos). Entrega LINK al video en un archivo README junto con el código del proyecto.

Funcionamiento del programa

El programa hace lo que se propone. Ejecuta correctamente la lógica del juego. No es trivial.

Evaluación por Paradigmas

Parte 1: Paradigma Procedural	25%
<ul style="list-style-type: none">• Funciones bien definidas con propósito claro, favorecen modularidad.• Uso adecuado de estructuras de datos y control de flujo (loops, condicionales).• Buen manejo del lenguaje para este paradigma.• Ejemplo: lógica de turnos y validación de jugadas.	✓
Parte 2: Paradigma Orientado a Objetos	25%
<ul style="list-style-type: none">• Clases bien diseñadas, representan entidades del problema con atributos y métodos relevantes.	✓

<ul style="list-style-type: none"> • Encapsulamiento correcto y uso de getters/setters donde sea necesario. • Uso eficiente de colecciones (List, Map, etc.). • Ejemplo: modelado del estado del juego, tablero, jugador, gato. 	
Parte 3: Paradigma Funcional	25%
<ul style="list-style-type: none"> • Uso de funciones puras, sin efectos secundarios ni variables globales. • Se mantiene la inmutabilidad de los datos donde sea posible. • Uso adecuado de recursión y funciones de orden superior (map, reduce, filter, etc.). • Ejemplo: estrategia de movimiento del gato con Streams. 	✓

Aspectos adicionales

- **Formato del código:** Se penaliza por indentación incorrecta o ausencia de comentarios clave. (-1 a -10 décimas)
- **Entrega fuera de plazo:** Se penaliza con -10 décimas por día de atraso.