

Tarea 4 – Conecta 4

Curso: CC1001 - Computación I – Sección 2

Profesor: José A. Pino

Profesores Auxiliares: Fernando Morales, Nicolás Pinilla

El popular juego *Connect Four* (distribuido mundialmente por Milton Bradley) consiste en dos jugadores que, alternando turnos, van colocando fichas de colores en un tablero vertical. En el juego gana quien consigue alinear primero 4 fichas en forma horizontal, vertical o diagonal.

En esta tarea se le pide que programe el juego. Se mantendrá la regla que solo pueden participar 2 jugadores, siempre alternando turnos. Ahora bien, dado que está implementando una versión electrónica del juego, se le pide que considere dos modos distintos a seleccionar al inicio de una sesión: jugador humano contra jugador humano y jugador humano contra computador.

TRABAJO PEDIDO

- 1) Diseñe un diálogo con el usuario que le permita obtener vía teclado qué tipo de juego será.

Por ejemplo:

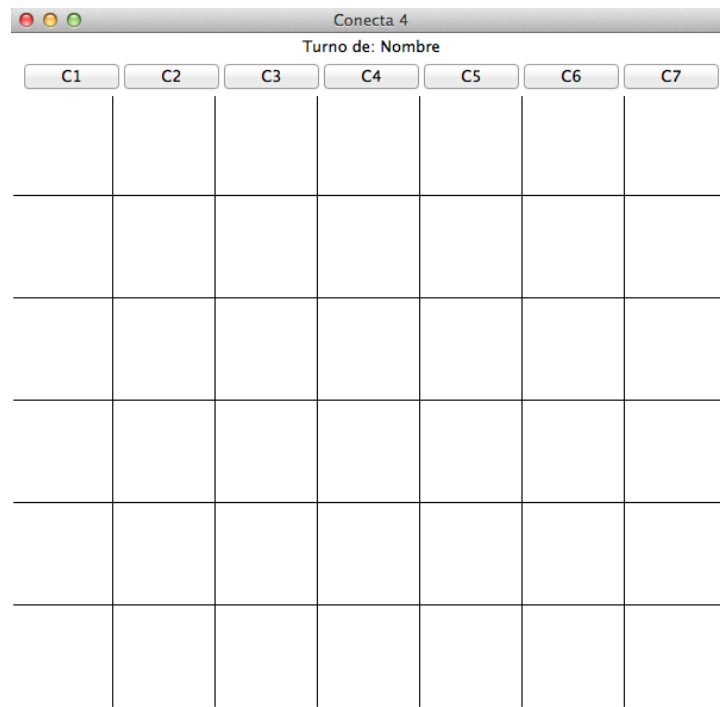
Bienvenido a Conecta 4!

Ingrese Modo de juego (1: humano-humano; 2: humano-computador): 1

Nombre Jugador 1: Nicolás

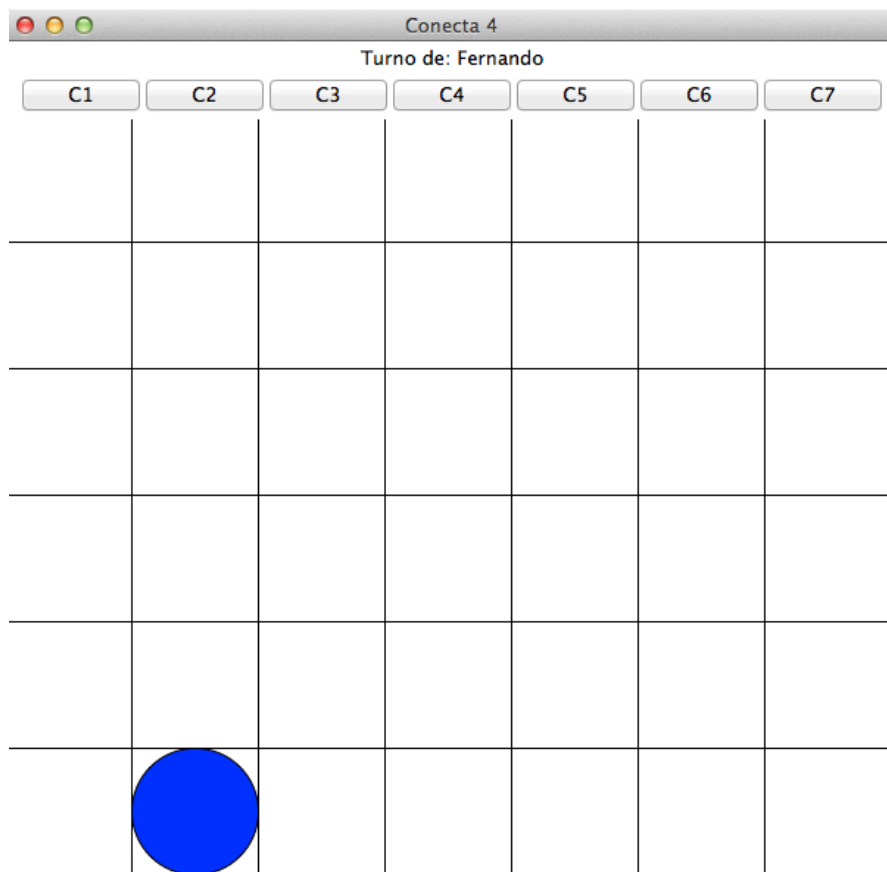
Nombre Jugador 2: Fernando

- 2) El tablero se les entrega listo en el archivo “`ventana.py`”

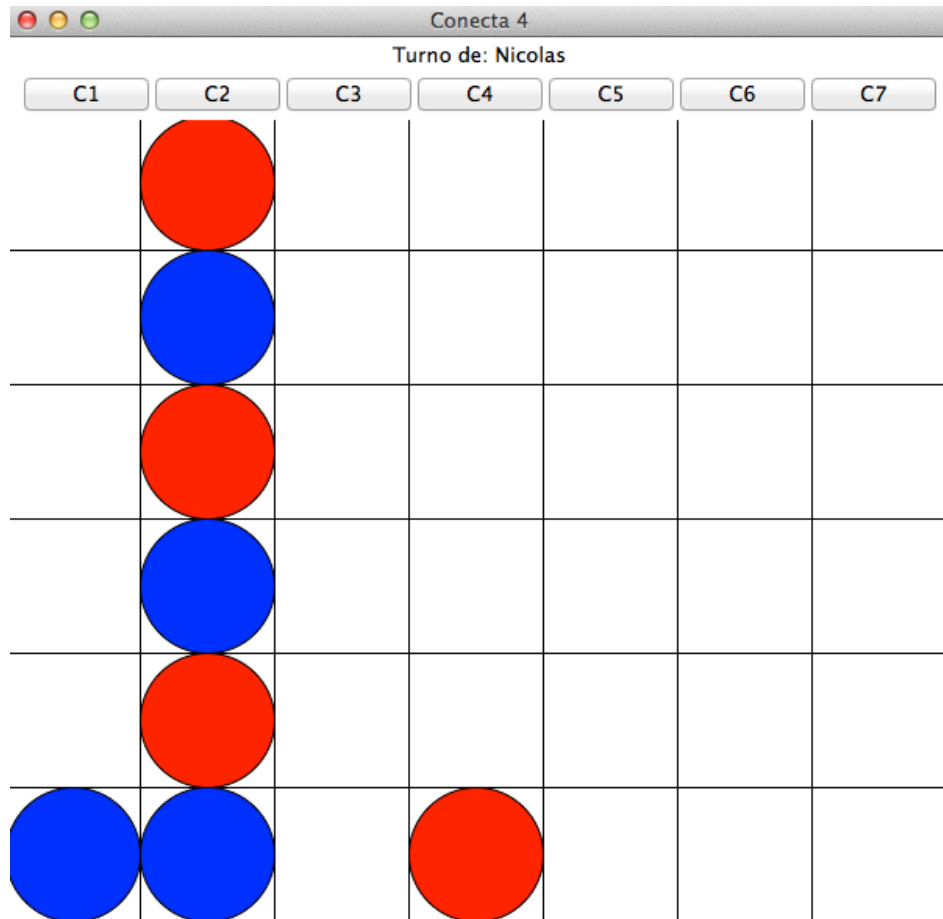


- 3) Represente internamente el tablero de juego como una tabla (lista de listas) de valores del tipo que usted decida. Si lo desea, para simplificar las siguientes operaciones, puede asociar a cada jugador un diccionario con las jugadas realizadas.
- 4) Implemente las funciones asociadas a cada uno de los botones. Al presionarlos, se debe dibujar una ficha en la primera fila disponible (de abajo hacia arriba) en la columna seleccionada. Si una columna ya no puede contener más fichas, se debe desactivar el botón respectivo. Esto lo puede conseguir con el comando: `boton.config(state=DISABLED)`. Además, debe actualizar internamente el estado del tablero de juego y decidir si la jugada efectuada fue o no ganadora. Finalmente, debe intercambiar de jugador e indicarlo modificando el texto en la etiqueta correspondiente. En el caso de estar jugando en el modo humano-computador, en el turno del computador se decide aleatoriamente la columna en la cual se debe agregar la siguiente ficha.

Por ejemplo, en el tablero inicial si se presiona el botón C2 se obtiene:



Y luego de un rato al presionar el botón C2 nuevamente:



BONUS (NO ES OBLIGATORIO ENTREGARLO)

Es claro que el computador, al elegir la columna de manera aleatoria, no está jugando de una manera muy inteligente. De hecho, uno esperaría tener un adversario a la altura, tal que siguiera una estrategia competitiva para intentar ganar.

Implemente las siguientes reglas simples que permitirían al computador, según diversos estudios sobre la matemática del *Conecta 4*, decidir cuál es la “mejor jugada” posible:

- Estrategia Defensiva: En primer lugar, se debe bloquear al adversario en el caso que pueda ganar en el próximo turno. Para ello, se debe examinar el conjunto de posibles próximas jugadas y, en el caso de detectar una posible línea (anticipando la jugada del adversario), el computador debe colocar una ficha con el cuidado de no volver a generar una combinación que permita ganar inmediatamente al otro jugador (por ejemplo, en una diagonal).

– Estrategia Ofensiva: Si no se da el caso anterior, por cada columna disponible se debe contar cuántas líneas ganadoras verticales, horizontales y diagonales es posible formar agregando una ficha en esa posición. La columna a elegir es aquella que maximiza el número de combinaciones ganadoras posibles.

INFORME

Junto con sus archivos Python, debe entregar un informe en formato PDF que documente su razonamiento e ilustre el correcto funcionamiento de su programa. Para ello, se le sugiere seguir la siguiente pauta:

- 1) Diseño de la solución: indicar la estrategia general que adoptó para resolver el problema presentado en la tarea, explicando cómo resolvió cada una de sus partes, al igual que las dificultades que encontró durante el desarrollo de cada una de ellas. Además, dé una lista con las funciones que diseñó para resolver el problema (cómo se llaman, qué parámetros reciben, qué parámetros devuelven, qué es lo que hacen). Note que NO SE LE PIDE NI DEBE COPIAR CÓDIGO FUENTE, sino que sea capaz de explicarlo con sus propias palabras.
- 2) Ejemplos de prueba: ejecute su programa en al menos tres casos distintos y representativos, considerando tanto situaciones en las que espera que funcione, como otras en las que espera que se caiga. Por ejemplo, muestre explícitamente cómo reacciona su programa al indicar valores nominales (aquellos en los que usted está seguro que su programa va a funcionar) y límites (como al dividir intencional o accidentalmente por 0, trabajar con valores fuera de rango, entre otros). Para cada caso, explique cuál es la situación que está probando, y muestre que su programa es lo suficientemente robusto, indicando cuál es el valor esperado y cuál es la respuesta que obtiene.
- 3) Experiencia: explique brevemente qué fue lo que realmente aprendió desarrollando la tarea y su apreciación personal frente a la resolución del problema.