



IIC1103 – Introducción a la Programación
1 - 2017

Enunciado Tarea 2

Recordatorio:

- **Fecha de entrega:** Miércoles 17 de mayo de 2017, a las 23:50 hrs.
- **Foro de consulta:** <https://goo.gl/g6LrzI>
- Este trabajo es **estrictamente personal**. Recuerda leer la Política de Integridad Académica del DCC disponible en <http://www.ing.uc.cl/ciencia-de-la-computacion/programas/licenciatura/politica-de-integridad-academica/>. Se usará un software **anti-plagio** para detectar similitud entre códigos.

¡Atención!

Ten en consideración que **no se recibirán entregas fuera del plazo**. Tampoco se te evaluará si te equivocas al entregar el archivo.

Es tu responsabilidad ir subiendo entregas parciales de tu tarea. Se revisará la última versión que hayas subido. Revisa la sección **Entrega** para las instrucciones de entrega de tu trabajo.

Objetivo

En esta tarea se espera que apliques en la solución de un problema todos los contenidos vistos en clases. Esto incluye listas, archivos y programación orientada a objetos, además de los contenidos de la Tarea 1.

El juego

Para esta tarea deberás implementar una variación del juego **Pentago**. A continuación, se explican las reglas del juego.

El juego original¹ consiste en un tablero de 6x6 que está dividido en 4 cuadrantes de 3x3. Sobre este, dos jugadores (representados por un color cada uno) deben colocar por turnos una bolita de su color en alguna posición del tablero y luego girar en 90 grados algún cuadrante (en cualquier sentido: dextrógiro² o levógiro³). El jugador que complete una línea recta de 5 bolitas en cualquier dirección (vertical, horizontal o diagonal) se convierte en el ganador. Si no quedan más espacios vacíos en el tablero y no ha habido ningún ganador, entonces se declara empate.

¹El juego lo pueden jugar en <http://perfect-pentago.net/>

²<http://dle.rae.es/srv/fetch?id=DcbkSfT>

³<http://dle.rae.es/srv/fetch?id=NCs528H>



Para tu tarea deberás implementar una generalización de este juego, para M participantes, sobre un tablero de tamaño $N \times N$ con cuadrantes de 3×3 , donde N es el número de espacios que será un múltiplo de 3.

Reglas del juego

- **Partida:** El juego comienza con un tablero vacío y se decide aleatoriamente el orden de los turnos de los jugadores.
- **Desarrollo:** El juego consiste en turnos intercalados que están divididos en dos fases:
 - **Fase de Colocación:** El jugador pone su bolita en una posición libre del tablero.
 - **Fase de Giro:** El jugador puede girar cualquier cuadrante del tablero independientemente de la posición en que colocó su bolita en la fase anterior.

Al comienzo del juego existirán cuadrantes neutros. Mientras sigan existiendo cuadrantes neutros, los jugadores podrán ignorar la fase de *Giro* en sus turnos. Un cuadrante neutro es aquel que está vacío o posee una sola bolita al centro del bloque, el que al ser girado no causa ningún efecto en el juego.

- **Término:** El juego termina cuando exista una fila ganadora en dirección vertical, horizontal o diagonal. Otro posible desenlace del juego es que no queden posiciones libres en el tablero, lo que genera un empate.

Archivos

Además de lo anterior, deberás ser capaz de leer y escribir archivos de estado de partidas que poseen el siguiente formato:

```
N,M,J
NombreJugador1
NombreJugador2
.
.
NombreJugadorM
0,0,0,0,0,0
0,0,0,0,0,0
```

0,0,0,0,0,0
0,0,0,0,0,0
0,0,0,0,0,0
0,0,0,0,0,0

A continuación se explican los parámetros del formato descrito:

- N: Tamaño del tablero. N siempre es un múltiplo de 3.
- M: Número de jugadores. Asume que M siempre es mayor o igual a 2.
- J: Es el número del último turno jugado.
- Jugadores: Se encuentran M líneas de jugadores que representan el nombre de los jugadores de la partida.
- Tablero: N líneas que contienen los datos del tablero. En este caso $N = 6$.
- Fila: Cada línea del archivo representa una fila en el tablero y se encuentra separada por comas. La cantidad de elementos en la fila es igual a N.
- Casilla: Cada casilla tiene $M+1$ posibilidades, donde el valor 0 es la casilla vacía. Para representar a los jugadores se tiene $\{1 \dots M\}$. Asume que no habrán valores fuera de estas $M+1$ posibilidades.

Además se adjunta un tablero de ejemplo, que es equivalente a la foto adjunta en la tarea para que puedas probar si funciona la lectura.

6,2,14
Jugador1
Jugador2
0,0,0,0,0,0
0,1,0,1,2,2
0,2,1,1,1,2
0,1,0,2,2,0
1,0,2,0,1,0
0,0,0,2,0,0

Interacción

Debes implementar un programa que permita a los jugadores participar por turnos intercalados. En particular el juego debería:

1. Poder importar una partida anterior.
2. Comenzar una nueva partida.
3. Pedir el nombre de los jugadores.
4. Entregar el orden de los turnos una vez, entregando el nombre de todos los jugadores.
5. Antes de cada turno, debe poder preguntarle al usuario si quiere exportar la partida.
6. Turno:
 - Mostrar nombre del jugador - Número de turno.
 - Mostrar estado actual del tablero.

- Recibir posición de la bolita a colocar.
- Mostrar estado actual del tablero.
- Recibir bloque que se desea girar y su giro.

7. Retornar jugador ganador una vez terminada la partida.

Restricciones

- Debe existir la clase `Tablero`.
- Se debe utilizar esta clase en la solución del problema.

Entrega

Debes guardar tu tarea en un archivo de nombre `tarea2_numero_alumno.py`, donde debes reemplazar `numero_alumno` con tu número de alumno. Por ejemplo, si tu número de alumno es 12345678 el nombre de la tarea sería `tarea2_12345678.py`. **Deberás crear un archivo .zip donde deberás incluir los archivos .py utilizados en tu tarea y deberá tener el nombre `t2_numero_alumno.zip`**

La entrega se realiza a través de un cuestionario en el Siding disponible en la página web del curso hasta el Miércoles 17 de mayo de 2017 a las 23:50 hrs. No se recibirán entregas atrasadas ni entregadas por otro medio.

Tareas que no sean entregadas en un archivo .zip, tendrán un descuento por formato de entrega de 0.5 pts a su nota final.