



Pontificia Universidad Católica de Chile
Escuela de Ingeniería
Departamento de Ciencia de la Computación
Introducción a la Programación – Primer Semestre 2015

Tarea 3: Juguemos Dominó

Objetivo General

En esta tarea deberás aplicar los conceptos de Orientación a Objetos para implementar un juego de Dominó entre 4 jugadores (<http://es.wikipedia.org/wiki/Dominio>).

Enunciado

Debes implementar un programa en Python que permita jugar Dominó entre 4 jugadores, utilizando objetos. El usuario de tu programa será el jugador 1 y el computador simulará a los otros tres jugadores. El usuario (jugador 1) juega en pareja con el jugador 3 (simulado), mientras que los jugadores (simulados) 2 y 4 juegan como la pareja rival. El usuario tiene libertad para elegir sus jugadas, pero tu programa deberá revisar que sean válidas.

El Dominó se juega normalmente utilizando 28 fichas, cada una de ellas con 2 números entre 0 y 6 (ver Fig. 1). El valor de una ficha se obtiene sumando sus dos números. Por ejemplo, la ficha (4,5) es de mayor valor que (6,2). La ficha (6,0) es de menor valor que la ficha (3,4).

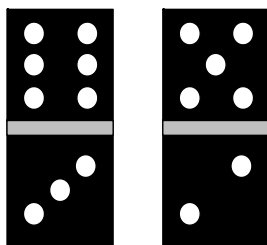


Figura 1. Las fichas (6,3) y (5,2). Las fichas son reversibles: estas dos podrían también leerse como (3,6) o (2,5). Existe una sola ficha con cada combinación de números (por ejemplo, existe una sola ficha con los números 3 y 6, y ésta puede ser representada como (3,6) ó (6,3)).

Al comienzo del juego se deben **generar** todas las **fichas**, **revolverlas** y ponerlas en una **pila** de fichas, y luego repartir 7 fichas a cada **jugador** (no quedan fichas libres). En consecuencia, cada jugador queda con una **mano** de 7 fichas. El jugador 1 (usuario) comienza la partida, poniendo en la **mesa** la primera ficha que elige de entre las que tiene en su mano. A continuación juega el jugador 2 (simulado por el PC), escogiendo de entre sus fichas (en su mano) alguna que sea válida para jugar.

Por ejemplo, si el usuario puso en la mesa la ficha inicial (6,4), el jugador 2 solo podrá jugar una ficha que tenga un 4 o un 6. Si juega una ficha con 6, la deberá ubicar al comienzo/cabeza de la mesa; si juega con una ficha con 4, deberá ponerla a la cola/final de la mesa. Después de cada jugada, tu programa deberá imprimir el estado de la mesa. Por ejemplo, si el jugador 2 jugó la ficha (6,2), se debiera imprimir la mesa:

--[2 | 6] -- [6 | 4] --

Figura 2. Ejemplo de impresión de la mesa,
después de la segunda jugada.

Nota que deben quedar los dos números 6 seguidos, dado que las fichas son reversibles. Si la ficha se había generado inicialmente como (6,2), se debe “invertir” y poner como (2,6).

Los jugadores siguen jugando en turnos: después del 2 juega el 3 y luego el 4. Cuando han jugado los 4 jugadores, se completa una ronda y se deberá comenzar la siguiente, partiendo nuevamente por el jugador 1 (el usuario). Un jugador puede pasar una jugada (no poner una ficha en la mesa) cuando no tiene fichas con los números de la cabeza o cola de la mesa. En el ejemplo anterior, después de la jugada, si el jugador 3 no tiene fichas con 2 o 4, deberá pasar.

El juego termina cuando uno de los jugadores se queda sin fichas o cuando los cuatro jugadores han pasado en forma sucesiva en una misma ronda. Gana la pareja del jugador que quedó sin fichas. Pero si el término es porque todos pasan, gana la pareja que obtiene la menor suma de valores en las fichas en sus manos.

Tu programa

Tu programa debe definir al menos las siguientes clases de objetos, con las características que se describen a continuación (puedes agregar todas las clases y métodos que quieras, acá solo se especifican los mínimos):

Clase	Atributos	Métodos
Ficha	Dos números (entre 0 y 6 cada uno)	valor() """Retorna el valor de la ficha resultante de la suma de sus dos números.""" rotar() """Retorna la ficha invertida"""
Pila	28 fichas de dominó en una lista desordenada (se recomienda usar random.shuffle())	sacar_ficha() """Retorna una ficha al azar que es tomada de las fichas que están en la pila. Además saca la ficha de la pila."""
Mesa	Las fichas que se van poniendo (ordenadamente) sobre la mesa	poner() """Pone correctamente una ficha válida en la cabeza o en la cola de mesa.""" imprimir() """Imprime en la pantalla las fichas que hay sobre la mesa."""
Jugador	Nombre del jugador (ingresado por el usuario o generado por el computador). Mano (conjunto de fichas en la mano del jugador)	buscar() """Busca en su mano una ficha válida para jugar. Si es simulado, retorna la ficha válida de mayor valor. Si es usuario, realiza el diálogo de input.""" jugar() """Pone una ficha en la mesa y la elimina de su mano.""" valor_mano() """Retorna la suma del valor de todas las fichas en su mano."""

El **programa principal** deberá:

- Crear una instancia de la clase **Pila** y otra de **Mesa**, y cuatro instancias de la clase **Jugador**.
- Preguntar el nombre del usuario e indicarle el nombre (simulado) de los jugadores rivales y de su pareja.
- Repartir las siete fichas a cada jugador.
- Mostrar al usuario la mano que recibió al inicio del juego (las fichas que tiene en su mano).
- Pedir al usuario que realice una jugada. (El usuario siempre pone la primera ficha en la mesa.)
- Hacer jugar al resto de los jugadores en turnos. Debe mostrar el estado de la mesa cada vez que se realiza una jugada. Cuando le toque jugar al usuario, muestra su mano antes de pedir su jugada.
- Detectar correctamente el término de una partida e indicar la pareja ganadora.
- Solo se debe permitir jugar fichas válidas y que el jugador efectivamente tenga en su mano.
- Por ahora, la única *inteligencia* de los jugadores simulados será la de poner la ficha válida de mayor valor en la mesa (para disminuir el valor de lo que queda en su mano).
- OPCIONAL
 - Hacer una clase **Partida**, que realice el juego del Dominó en rondas sucesivas (hasta que todos pasan) y juegos en turnos dentro de cada ronda.
 - Agregar más inteligencia a los jugadores artificiales.

Recomendación

Para desarrollar tu programa, te recomendamos un desarrollo en “prototipos evolutivos”. Primero haces (en papel) una definición gruesa de las clases y el algoritmo de tu programa principal que activará los objetos en una primera ronda. Luego escribes tu programa Python solo para la primera ronda: haces el diálogo con el usuario (para recibir su jugada “válida”), y le agregas impresiones en diversos lugares para asegurar que se están generando adecuadamente las 28 fichas, se están revolviendo y se están entregando 7 fichas al azar a cada jugador. Así puedes revisar si la jugada que se pone en la mesa está correcta. Una vez que esta primera ronda está funcionando bien, le agregas un ciclo para hacer todas las rondas necesarias hasta que el juego termine.

Para imprimir los atributos de un objeto, se recomienda utilizar el método especial `__str__`. No es necesaria una impresión gráfica sofisticada como la de la Figura 1. Basta con algo parecido al ejemplo de la Figura 2.

Entrega

Esta tarea no tiene nota, pero será evaluada a través de un control el día **sábado 6 de Junio**. El control evaluará todas las habilidades y conocimientos que esperamos adquieras mediante la realización de esta tarea.

Al igual que en la tarea anterior, deberás entregar semanalmente (los días viernes) un avance a través del SIDING.