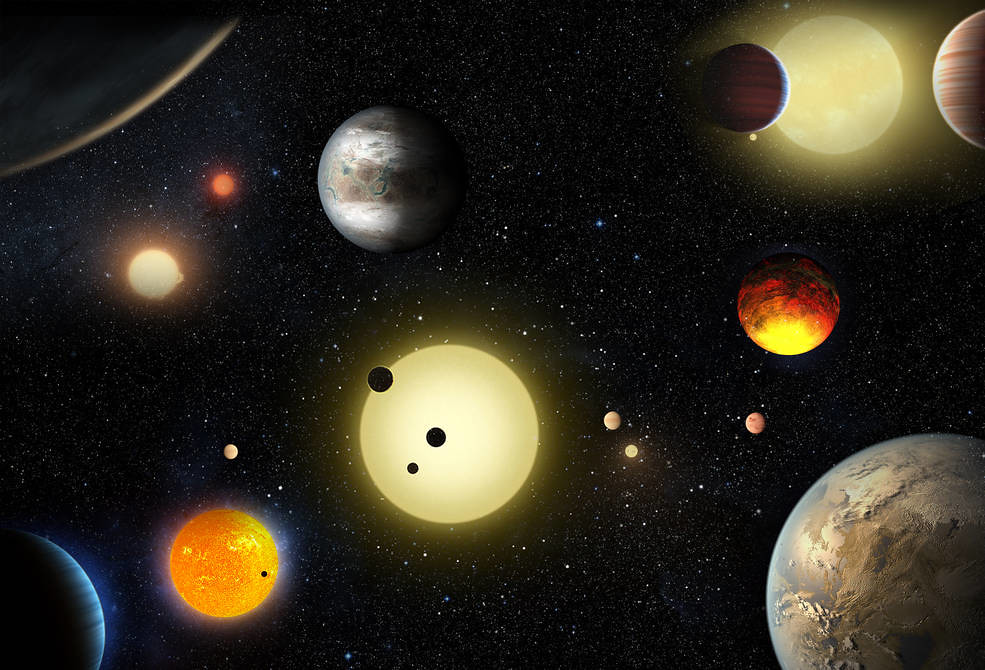
** T.P Final CTEDyA.**

Nombre: Bruno Morales.

Profesor: [Leonardo Javier AMET](https://campus.unaj.edu.ar/user/view.php?id=4737&course=2875)

Comisión 5.

Fecha: 03/12/2020



**Introducción**

El juego consiste en atacar la mayor cantidad de planetas posibles, para derrotar al rival, en este caso el rival es un bot, que va a intentar hacer lo mismo con el jugador humano.

El bot se distingue por el color azul, mientras que el jugador humano es el rojo, y los planetas neutrales de color blanco.

Cada jugador comienza en una posición a azar con un solo planeta y con una cantidad aproximada de 30 tripulantes. Cuando se ataca se envía la mitad de tripulantes del planeta en el que se encuentra, y esos tripulantes pasan a descontarse del planeta que es atacado. Por ejemplo, si el bot tiene 30 planetas, y ataca a un planeta que tiene 5 tripulantes rivales, cuando envía los 15 para atacar, esos 10 tripulantes que quedarían después del ataque (5 – 15), pasarían a ser del atacante.

Cuando se conquiste un planeta se empezará aumentar su población. Los únicos planetas que no se incrementan son los neutrales.

Quien logre atacar la mayor cantidad de planetas y vencer todos los planetas del rival, es el ganador.

**Detalles de implementación:**

El programa se construye además de la clase “Estrategia”, de otras 5 clases creadas a partir del contenido visto en la materia, con la implementación de algunos de los recorridos de ArbolGeneral, como PreOrden, o PorNiveles adaptados a la búsqueda de Planetas.

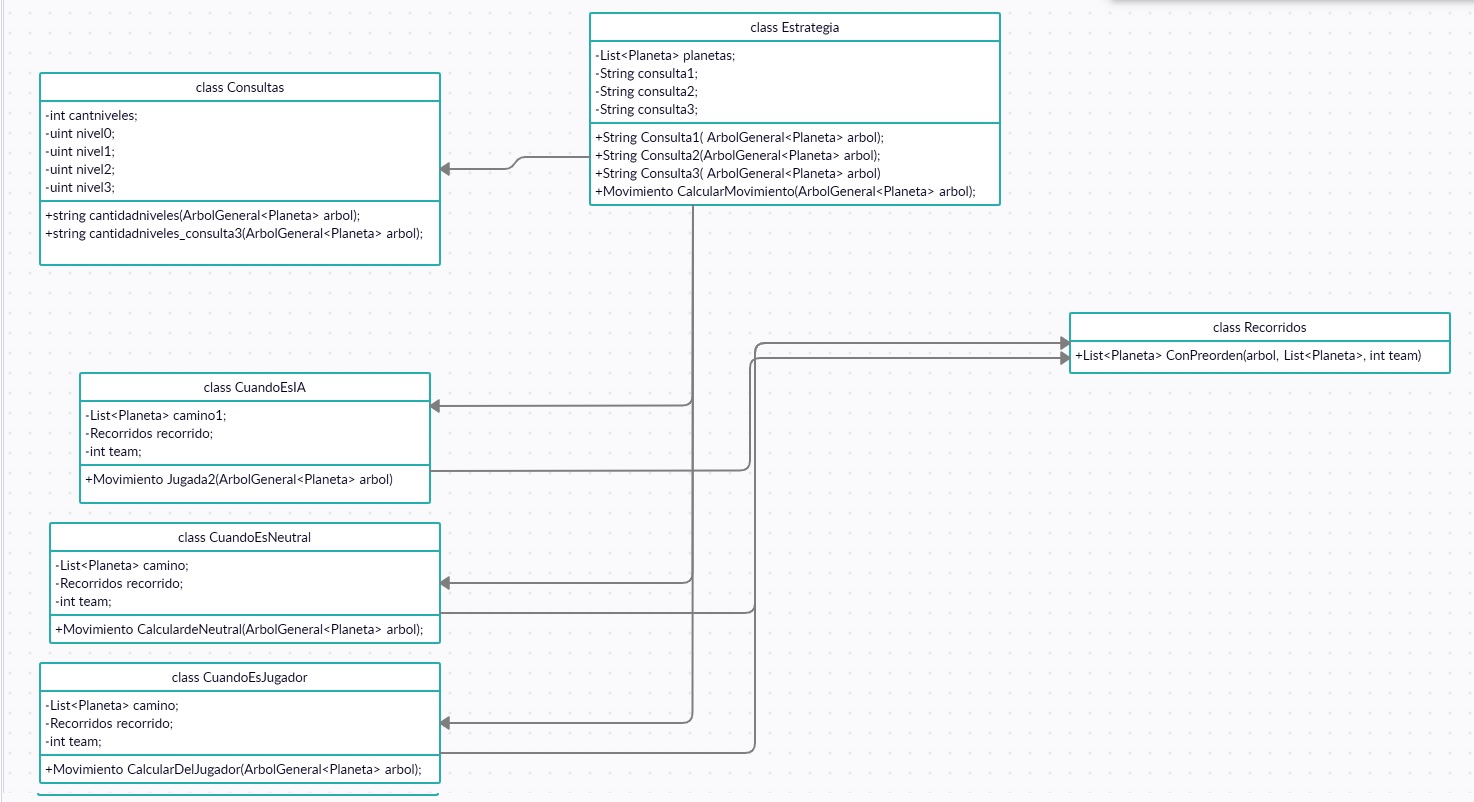
Para la primera consulta, se crea una “lista” de planetas, se llama a la clase “Recorridos” que contiene el recorrido, en este caso el PreOrden, y guarda en un String la cantidad de planetas que existen hasta el bot.

Para la Segunda consulta se llama de la clase “Consultas” al método “cantidadniveles” que hace un recorrido por niveles, sumando a variables como por ejemplo “nivel1”, cada vez que la población del Planeta sea mayor a 10.

La Tercera consulta, es similar a la segunda, pero con la variable que guarda la cantidad de población de un nivel, y la cantidad de planetas por otro. Al final se realiza la operación de dividir la suma total de planetas, con la cantidad de planetas, por niveles.

Para el método “CalcularMovimiento” se realiza la estrategia de chequear a quién pertenece la “raíz”. Si es un planeta Neutral, se llama a la clase “CuandoEsNeutral” que realiza un recorrido PreOrden, y guarda en una Lista, el recorrido desde la Raíz hasta el Bot o Planeta de la IA.  
Una vez con el recorrido, para calcular el movimiento, se establece como “Planeta Origen” el ultimo planeta, después se elimina de la lista con un Remove, y se establece el planeta que queda, esta vez último, como el “Planeta Destino”. Cuando la Raíz es del jugador, se realiza el mismo procedimiento que cuando es planeta neutral. Cuando la raíz es del bot, realiza un recorrido similar, pero para establecer el ataque, revisa de quién es el planeta que le sigue a la raíz, y así realiza el movimiento.

**Diagrama de UML :**



**Imagen de las clases :**

****

****

****

****

****

****

**Una breve conclusión o reflexión de la experiencia adquirida a partir de la realización del trabajo final.**

La conclusión no es del “T.P Final” sino de la materia y de la cursada en sí.  
Se aprendió muchísimo en cuanto a conceptos de estructuras con una claridad a la hora de las explicaciones, que no había notado en ninguna otra materia. Los contenidos de estructuras de datos, se me ocurre que se pueden aplicar a distintas funcionalidades con otros objetivos. En la materia además de abordar los contenidos propios de la materia, también se aprendió a analizar el código, a mirarlo y ver parte por parte cuando ocurren errores, que es fundamental para buscar por nosotros mismos, e incluso buscar definiciones de cómo funcionan palabras, métodos, palabras reservadas o de cosas que no están dentro de la materia pero que sirven un montón para entender como se programa, y como se busca soluciones reales a los problemas de la programación.

Agradezco enormemente la clara explicación del profesor [Leonardo Javier AMET](https://campus.unaj.edu.ar/user/view.php?id=4737&course=2875) en todos los temas de la materia, y por la enseñanza de como analizar, mejorar y solucionar problemas de código.