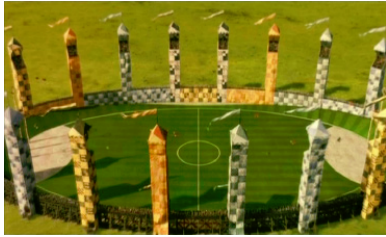


Picadito de Quidditch



El Quidditch es un deporte que juegan los magos y brujas de todo el mundo, donde se enfrentan 2 equipos volando con escobas mágicas en un campo con forma oval que en cada uno de los lados, uno para cada equipo, está dotado de tres aros ubicados a un distinto nivel de altura. Cada equipo está conformado por: un guardián, dos golpeadores, tres cazadores y un buscador.

Para jugar se utilizan 3 tipos de pelotas distintas:

- Quaffle: Los cazadores deben meter la quaffle en alguno de los aros del equipo oponente para sumar a su equipo 10 puntos. El guardián debe proteger los aros y evitar que le metan un gol. Hay sólo una en el campo de juego.
- Bludger: Las bludgers son pelotas que aportan emoción (y violencia) a los partidos; los golpeadores pueden batear una bludger apuntando hacia un jugador del otro equipo y los más grosos hasta pueden evitar goles desviando la quaffle. Hay 2 bludgers en un partido.
- Snitch: El buscador es el encargado de buscar la snitch y atraparla; al hacerlo suma 150 puntos para su equipo y termina el partido.

Queremos armar un sistema usando el paradigma orientado a objetos para modelar este extravagante juego. Se pide testeo, diagrama de clases y código para:

1)

- a. Saber el **nivel de manejo de escoba** de un jugador. Esto se calcula como los skills del jugador / su peso.
- b. Saber la **velocidad** de un jugador, que es la velocidad de la escoba que use multiplicado por su nivel de manejo de escoba. Actualmente existen estas escobas:
 - Nimbus: La velocidad se calcula como $80 - \text{la cantidad de años desde su fabricación}^1$ por el porcentaje de salud de la misma. Por ejemplo, una Nimbus 2001 con un 50% de salud tendría una velocidad de $(80-22)*0.5 = 29$ km/h.
 - Saeta de Fuego: es la escoba más veloz fabricada, la velocidad es de 100 km/h.
- c. Saber la **habilidad** de un jugador. Como cada jugador tiene un entrenamiento acorde a su posición en el campo, su habilidad se calcula de formas diferentes:
 - Cazadores: su velocidad + sus skills + su puntería * su fuerza.
 - Guardianes: su velocidad + sus skills + su nivel de reflejos + su fuerza.
 - Golpeadores: su velocidad + sus skills + su puntería + su fuerza.
 - Buscadores: su velocidad + sus skills + su nivel de reflejos * nivel de visión.

2) Se desea saber..

- a. Si un jugador **le pasa el trapo** a otro; esto sucede si es por lo menos el doble de habilidoso que el otro jugador.
- b. Si un jugador **es groso**, que se cumple si su habilidad es mayor al promedio de su equipo y su velocidad mayor a un valor arbitrario que a medida que el mercado de escobas mejora se actualiza para todos por igual.
- c. Si un equipo **tiene un jugador estrella** para jugar contra otro equipo. Un jugador es estrella si le pasa el trapo a todos los jugadores del equipo contrario.

3) Hacer **que un equipo juegue contra otro**. Cuando esto pasa cada jugador del equipo juega un

¹ `new Date().year()` devuelve el año actual.

turno de acuerdo a su puesto, como se describe a continuación...

Nota: Todas las colecciones entienden un mensaje `anyOne()` y retornan un elemento al azar. Los intervalos de números (rangos) son colecciones que pueden crearse haciendo por ejemplo `(1..10)`

El **cazador** cuando juega, si tiene la quaffle debe intentar meter gol, que implica:

- Evitar bloqueos: cada uno de los jugadores contrarios **intenta bloquear** el tiro del cazador, y en caso de **poder bloquearlo** (ver 4.a) se interrumpe el tiro. Lógicamente en cuanto uno bloquea, los jugadores restantes no deben seguir bloqueando.
- El equipo gana 10 puntos por meter gol.
- El cazador gana 5 puntos de skills

Si el tiro se interrumpe por un bloqueo, el cazador pierde 2 puntos de skills y el que bloqueó gana 10. Independientemente de si pudo meter gol o si fue bloqueado, el cazador **pierde la quaffle**.

Siempre que un cazador pierde la quaffle, ésta es atrapada por el cazador rival más rápido.

El **buscador** cuando juega intenta obtener la snitch. Cuando comienza el partido los buscadores arrancan buscando la snitch, si la encuentran deben perseguirla hasta atraparla.

Si el buscador está **buscando la snitch**, debe hacer un random entre 1 y 1.000 y si el número obtenido es menor a su habilidad + la cantidad de turnos continuos buscando entonces encontró la snitch.

Si está **persiguiendo la snitch**, debe recorrer 5.000 kms para poder atraparla. En cada turno recorre una cantidad de kms igual a su velocidad / 1,6. Una vez que la atrapó, aumenta sus skills 10 puntos y su equipo gana 150 puntos.

El **golpeador** cuando juega elige un **blanco útil** (ver 4.b) del equipo contrario al azar. Si puede golpear a su blanco, el mismo sufre los efectos de **ser golpeado por una bludger** (ver 4.c) y el golpeador sube sus skills en 1. Para poder golpear al otro debe cumplirse que la puntería del golpeador sea mayor que el nivel de reflejos del blanco o si saca por lo menos 8 en un random de 1 a 10. No pasa nada si pifia.

El **guardián** no hace nada en su turno, sólo participa activamente cuando hay que bloquear.

4) Últimos detalles!

- a. Saber si un jugador **puede bloquear** el tiro de un cazador. Esto se cumple para los golpeadores si son grosos, para los guardianes si sacan 3 en un random de 1 a 3, y para los cazadores si le pasan el trapo al cazador que tiró. Los buscadores no bloquean.
- b. Saber si un jugador es un **blanco útil**. Se cumple para un cazador si tienen la quaffle, para un buscador si está buscando la snitch o le faltan menos de 1000 kilómetros y para un guardián si su equipo no tiene la quaffle. No es útil golpear golpeadores.
- c. Hacer que un jugador **sea golpeado por una bludger**. Cuando esto pasa el jugador pierde 2 puntos de skills y su escoba recibe un golpe (las Nimbus pierden 10% de salud, a las Saetas de Fuego no les pasa nada). Además los buscadores deben **reiniciar la búsqueda** y los cazadores, si tenían la quaffle, **la pierden**.

- 5) **BONUS!** Cuando un buscador groso es golpeado por una bludger, el mismo queda aturdido un turno y en el siguiente puede retomar su actividad tal cual estaba. Un buscador aturdido es blanco útil si la actividad a retomar lo llevaría a ser útil en el siguiente turno. **No cambiar lo hecho en el punto 3 para agregar esta lógica.**