

## Ejercicio de diseño: Expertos en maquinitas

Queremos desarrollar un programa en Haskell para modelar los juegos de un casino y la posibilidad de los jugadores de ganar o salir con los bolsillos vacíos.

La información que disponemos de las personas es su nombre, la cantidad de dinero que tienen, su suerte (como porcentaje de veces que suele ganar naturalmente) y una lista de factores que podrían ayudarles a ganar distintos juegos, como ser su inteligencia, algún amuleto, etc. De cada factor tenemos un valor numérico que nos indica qué tan valioso es.



Tenemos un par de personas de ejemplo para poder probar nuestro programa:

- Cosme Fulanito, tiene \$100, 30% de suerte y los siguientes factores: un “amuleto” de valor 3, “manos magicas” de valor 100 y “paciencia” de valor -10.
- El Coco, tiene \$20, 70% de suerte y los siguientes factores: “inteligencia” de valor 55 y “paciencia” de valor 50.

1. Modelar a las personas y declarar a **cosmeFulanito** y a **elCoco** en base a dicha definición.
2. Sobre las personas queremos saber:
  - a. Cuánto tiene para un determinado factor, retornando 0 si no lo tiene.
  - b. El potencial de una persona, que es su suerte más la sumatoria del valor de sus factores dividida por 100.
  - c. Si es tramposa, que será cierto si alguno de sus factores no está permitido en el casino, lo cual de ahora en más se indicará para cada factor.
3.
  - a. Modelar a los juegos sabiendo que un juego se compone por un nombre, una forma de determinar cuánto dinero da como ganancia ante una apuesta y una serie de criterios determinantes para poder ganarlo.
  - b. Saber si un jugador puede ganar un juego, lo cual sucede si cumple todas las condiciones para ganar ese juego.
  - c. Declarar la **ruleta** que da como ganancia 37 veces la apuesta. Para ganar la persona debe tener un potencial mayor a 70 y no ser tramposo.
  - d. Declarar la **maquinita** que se basa en un jackpot, y lo que se gana es ese jackpot + la apuesta. Para ganar se debe cumplir que la persona tenga una suerte mayor a 50 y que tenga más de 40 de paciencia.
  - e. Mostrar un ejemplo de uso de la función del punto b con una de las personas del punto 1 usando una ruleta y otro ejemplo usando una maquinita.
  - f. Declarar al **blackjack** que paga el doble de la apuesta. Para ganar la persona debe tener una suerte mayor a 30 o bien cumplir con ser tramposo.
4. Hacer que un jugador apueste en un juego, que implica que la persona baje su saldo en el valor de la apuesta y luego juegue al juego. Al jugar, si puede ganar en ese juego, la persona debería aumentar su saldo en lo que gana en el juego, de lo contrario la persona no gana nada. El sistema no debería permitir apostar un monto superior al saldo que tiene la persona.
5. Dado un jugador, una apuesta inicial y una lista de juegos, obtener la cantidad total de dinero que puede conseguir esa persona con ese monto si apuesta en cada juego lo conseguido en el juego anterior, evitando los juegos en los cuales no pueda ganar. Si no puede ganar en ningún juego, el resultado sería la apuesta inicial, ya que no apostaría en ninguno.