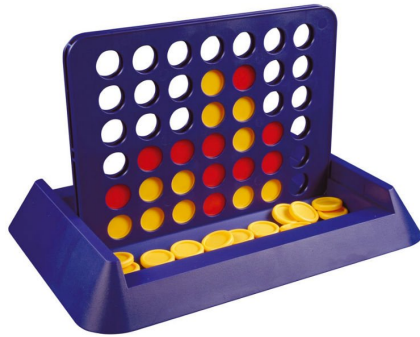


# Pràctica complementària de Haskell.

## Quatre en ratlla

### 1 Presentació

Considerem el conegut joc del “quatre en ratlla”. El tauler és de  $n \times m$ , on  $n$  i  $m$  es defineixen al iniciar la partida. El cas més habitual és el  $6 \times 7$ , és a dir, 6 files i 7 columnes. Cada jugador té  $\frac{n \times m}{2}$  fitxes d'un mateix color. Per exemple, unes son vermelles i les altres grogues.



Els jugadors introdueixen alternativament una fitxa del seu color en una columna. La fitxa cau fins dipositar-se al d'amunt de l'última fitxa que s'ha introduït en la mateixa columna.

Un jugador guanya quan després d'introduir una fitxa hi ha 4 fitxes consecutives del seu color formant una línia horitzontal, vertical o diagonal.

La partida acaba quan un jugador guanya o s'omple el tauler.

### 2 Es demana

Fer un programa Haskell que admeti dues formes de joc

1. L'usuari juga contra el programa.
2. Simular una partida del joc entre dos jugadors.

La primera opció llegeix els *moviments* de l'usuari (indicant cada cop la columna seleccionada) i dona els seus alternativament fins que acaba la partida. Sempre comença l'usuari. Al finalitzar, s'ha d'indicar el resultat: empat, guanya l'usuari o guanya el programa.

En la segona opció s'han de mostrar els moviments dels dos jugadors d'un cop i el resultat de la partida.

Per a definir la forma de jugar del programa s'usarà una *estratègia*, que és una funció que rep l'estat del tauler i el jugador a qui toca moure i retorna la columna on posar la fitxa. Heu de fer les funcions de joc parametritzades amb l'estratègia.

Una partida simulada necessita les dues estratègies que usarà el programa (una per jugador). Noteu que aquests tipus de partida no requereix cap interacció amb l'usuari.

Una partida contra l'usuari necessita l'estratègia que usarà el programa. En aquest cas s'ha d'usar l'entrada/sortida per introduir els moviments de l'usuari.

Cal que programeu diverses estratègies, com a mínim una d'aleatòria i una d'aleatòria però una mica intel·ligent. Es valorarà la qualitat de les estratègies.

Per obtenir nota d'aquesta part és indispensable implementar la simulació de partida. Per obtenir la màxima nota heu de fer els dos tipus de partida i que es puguin escollir les estratègies d'una llista (posant, per exemple, un número). En aquest darrer cas l'usuari triarà al principi el tipus de partida. Recordeu que en tots els casos s'ha d'introduir també la mida del tauler.

El codi s'ha de poder compilar i generar un executable.