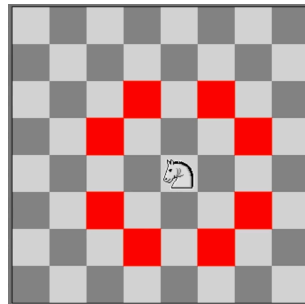


Considerem un cavall en un tauler de  $8 \times 8$  escacs buits. La seva posició es pot donar amb una tupla que indiqui la seva fila i la seva columna:

```
type Pos = (Int, Int)    -- la casella inferior esquerra és (1,1)
```

Recordeu que els cavalls es mouen en "L":



1. Definiu una funció `dins :: Pos → Bool` que, donada una posició d'un cavall, retorni si aquesta és dins del tauler.
2. Definiu una funció `moviments :: Pos → [Pos]` que, donada una posició d'un cavall dins del tauler, retorni la llista de posicions dins del tauler on es pot trobar després d'un salt.

L'ordre de la llista no és important: Els jocs de proves ja l'ordenen amb `sort`. Però heu d'escriure `import Data.List (sort)` al principi del vostre programa.

3. Definiu una funció `potAnar3 :: Pos → Pos → Bool` que, donada una posició inicial  $p$  dins del tauler i una posició final  $q$ , digui si un cavall pot anar de  $p$  a  $q$  en (exactament) tres salts.
4. Definiu ara una funció `potAnar3' :: Pos → Pos → Bool` que faci el mateix que `potAnar3` però trient partit del fet que les llistes són instància de **Monad**.

A l'hora de corregir es tindrà en compte la correcció, consició, senzillesa, claredat, elegància i l'ús de les eines de programació funcional de la solució proposada. En aquest problema, l'eficiència és poc important.

### Exemple d'entrada 1

```
dins (4, 5)
dins (0, 1)
dins (4, 9)
sort $ moviments (4, 5)
sort $ moviments (1, 1)
potAnar3 (1, 1) (4, 5)
potAnar3 (1, 1) (4, 6)
potAnar3' (1, 1) (4, 5)
potAnar3' (1, 1) (4, 6)
```

## Exemple de sortida 1

```
True
False
False
[(2, 4), (2, 6), (3, 3), (3, 7), (5, 3), (5, 7), (6, 4), (6, 6)]
[(2, 3), (3, 2)]
True
False
True
False
```

## Informació del problema

Autoria: Jordi Petit

Generació: 2026-02-03T17:08:11.672Z

© *Jutge.org*, 2006–2026.  
<https://jutge.org>