Bűvös négyzet

Feladat ismertetése

A bűvös négyzet egy olyan négyzetes ($n * n$) mátrix, amelynek minden sorában illetve oszlopában, valamint a fő- és a mellékátlójában található elemeinek összege ugyanannyi, nevezzük ezt varázskonstansnak.

$$\begin{bmatrix} 2 & 9 & 4 \\ 7 & 5 & 3 \\ 6 & 1 & 8 \end{bmatrix}$$

A fenti mátrix bűvös négyzet és a varázskonstansa 15, mert vegyünk példakénmat az első oszlopát, az első sorát és mindkét átlóját, ebben az esetben igaz az, hogy

$$2 + 7 + 6 = 2 + 9 + 4 = 2 + 5 + 8 = 4 + 5 + 6 = 15$$

Főátló

Egy négyzetes mátrix főátlójának azokat az elemeit nevezzük, amelyeknek a sor- és oszlopindexe megegyezik.

Például: $\begin{bmatrix} 2 & 3 & 1 \\ 7 & 6 & 2 \\ 0 & 0 & 0 \end{bmatrix} Főátló = 2 + 6 + 0 = 8$

Mellékátló

Egy négyzetes mátrix mellékátlója hasonlóan számolandó, mint a főátló, csak ebben az esetben pont jobb fentről indulunk.

Például: $\begin{bmatrix} 2 & 3 & 1 \\ 7 & 6 & 2 \\ 0 & 0 & 0 \end{bmatrix} Mellékátló = 1 + 6 + 0 = 7$

Konvenciók

Ahhoz, hogy megfelelően dolgozhassunk, egy beszédes típusszinoníma bevezetésére lesz szükségünk.

Kezdjünk hát ezzel:

```haskell
type Matrix = [[Int]]
```

Ezzel a jelöléssel a kezdeti bűvös négyzet a következőképpen írható fel:

```haskell
[[2,9,4], [7,5,3], [6,1,8]] :: Matrix
```

Első szint: Azonos méretű mátrixok (2 pont)

Döntsük el, hogy két mátrix egyforma méretű-e, vagyis ugyanannyi soruk és oszlopuk van. Feltehető, hogy a vizsgált mátrixok páronként vett azonos sora vagy oszlopa közül legalább az egyik véges.

```
isSameSize :: Matrix -> Matrix -> Bool
```

```
isSameSize [] []
isSameSize [[1,2]] [[5,6]]
isSameSize [[1,2,3]] [[5,6,7]]
isSameSize [[1,2,3], []] [[5,7,6], []]
isSameSize [[1,2,3], [5,5]] [[5,7,6], [6,7]]
not $ isSameSize [[1,2]] [[5]]
not $ isSameSize [[1,2]] [[5,6,7]]
not $ isSameSize [[1,2,3]] [[5,6,7,6]]
not $ isSameSize [[1,2,3], []] [[5,6,7,6], []]
not $ isSameSize [[1,2,3], []] [[5,7,6], [6]]
not $ isSameSize [[1,2,3], []] [[5,7,6], [6,7]]
not $ isSameSize [[1,2,3], [5,5]] [[5,7,6], [6,7,8]]
not $ isSameSize [[1,2,3], [5,5]] [[5,7,6], [6,7,8]]
not $ isSameSize [[1,2,3], [5,5]] [[5,7,6], [1,2..]]
not $ isSameSize [[1,2,3], [5,5]] [[5,6..], [1,2..]]
not $ isSameSize [[1..], [5,5]] [[5,6,7], [1,2..]]
```

Második szint: Bűvös négyzet (5 pont)

Döntsük el egy kapott mátrixról, hogy az egy bűvös négyzet-e, ha igen, adjuk meg a varázskonstansát. Ehhez nincs más dolgunk, mint megnézni, hogy ez egy négyzetes mátrix-e, illetve, hogy minden sorának, oszlopának és mindkét átlójának ugyanannyi az összege és minden szám különböző, úgy, hogy 1-től kezdődjenek. Definiáljuk a `magicSquare` függvényt, ami kap egy mátrixot és visszaadja a varázskonstansát `Just`-ba csomagolva, ha az bűvös négyzet, ha nem, akkor `Nothing`-al tér vissza.

```
magicSquare :: Matrix -> Maybe Int
```

```
magicSquare [[2,9,4], [7,5,3], [6,1,8]] == Just 15
magicSquare [[7, 12, 1, 14], [2, 13, 8, 11], [16,3, 10, 5], [9, 6, 15, 4]] == Just 34
magicSquare [[1,1,1], [1,1,1], [1,1,1]] == Nothing
magicSquare [[2,9,4], [7,5,3], [6,1,16]] == Nothing
magicSquare [[2,9,4], [7,0,3], [6,1,8]] == Nothing
magicSquare [[2,1,4], [7,0,3], [6,1,16]] == Nothing
magicSquare [[4,11,6], [9,7,5], [8,3,10]] == Nothing
magicSquare [[2,9,4], [7,5,3], [6,1,16]] == Nothing
magicSquare [[2,9,4], [7,0,3], [6,1,8]] == Nothing
magicSquare [[2,1,4], [7,0,3], [6,1,16]] == Nothing
magicSquare [[1,2,3], [4,5,6], [7,8,9]] == Nothing
magicSquare [[7, 12, 1, 14], [2, 13, 8, 11], [16, 3, 10, 5]] == Nothing
magicSquare [[1,2,5,6],[3,4,5,6],[8,9,10,11],[12,13,14,15]] == Nothing
magicSquare [[1,1], [2,2]] == Nothing
magicSquare [[1,2], [2,2]] == Nothing
```

```
magicSquare [[1]] == Just 1
magicSquare [[1,2],[3,4]] == Nothing
magicSquare [[7, 12, 1, 5, 14], [2, 13, 8, 11], [16,3, 10, 5], [9, 6, 15, 4]] == Nothing
magicSquare [[7, 12, 1, 14], [2, 13, 8, 11], [16, 3, 10, 5], [9, 6, 15, 9, 4]] == Nothing
```

Harmadik szint: Minimális költségű transzformáció (10 pont)

Valójában, az ezeket a feltételeket kielégítő $3 * 3$-as bűvös négyzetből csak egyetlen egy van, az összes többi az ő forgatásából és transzponálásából jön létre.

Transzponálás

A transzponálás, tulajdonképpen a főátlóra történő tükrözést jelent, vagyis az i-ik sorból i-ik oszlop lesz. Legyen A egy $m * n$-es mátrix. Legyen A^T $n * m$-es mátrix A transzponáltja úgy, hogy

$$A^T = [a_{ij}]^T = [a_{ji}]$$

Például: $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}^T = \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$

Forgatás

A forgatás alatt csak egyszerűen arra gondolunk, mintha a mátrix középső eleme körül elforgatnánk a többi elemet.

Például: $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \xrightarrow{\circlearrowright 90^\circ} \begin{bmatrix} 7 & 4 & 1 \\ 8 & 5 & 2 \\ 9 & 6 & 3 \end{bmatrix}$

Ezek alapján legenerálhatjuk az összes lehetséges elrendeződését a $3 * 3$-as bűvös négyzetnek, mert vagy bemutatott mátrixot forgatjuk el $0°$, $90°$, $180°$ és $270°$-al, vagy pedig a transzponáltját, így megkapjuk a `8` különböző elrendeződését.

Minimális költségű bűvös négyzet transzformáció

Írjunk egy olyan függvényt, amely ha kap egy $3 * 3$-as mátrixot, akkor meghatározza, hogy mennyi a minimális költsége annak az átalakításnak, amellyel a kapott mátrixunkat bűvös négyzetté alakíthatjuk. Amennyiben a kapott mátrix nem $3 * 3$-as, abban az esetben térjünk vissza `Nothing`-al.

Az átalakítási költségen egy számot értünk, amelyet úgy számolunk ki, hogy a két mátrix ugyanazon indexein szereplő elemei különbségének abszolútértékét összegezzük.

```
minimalCost :: Matrix -> Maybe Int
```

```
minimalCost [[2,9,4], [7,5,3], [6,1,8]] == Just 0
minimalCost [[7, 12, 1, 14], [2, 13, 8, 11], [16,3, 10, 5], [9, 6, 15, 4]] == Nothing
minimalCost [[1,1,1], [1,1,1], [1,1,1]] == Just 36
minimalCost [[2,9,4], [7,5,3], [6,1,16]] == Just 8
minimalCost [[2,9,4], [7,0,3], [6,1,8]] == Just 5
minimalCost [[4,11,6], [9,7,5], [8,3,10]] == Just 18
minimalCost [[4,9,8], [11,7,3], [6,5,10]] == Just 18
minimalCost [[8,9,4], [3,7,11], [10,5,6]] == Just 18
minimalCost [[2,9,4], [7,5,3], [6,1,16]] == Just 8
minimalCost [[2,9,4], [7,0,3], [6,1,8]] == Just 5
minimalCost [[2,1,4], [7,0,3], [6,1,16]] == Just 21
minimalCost [[1,2,3], [4,5,6], [7,8,9]] == Just 24
minimalCost [[7, 12, 1, 14], [2, 13, 8, 11], [16, 3, 10, 5]] == Nothing
minimalCost [[1,2,5,6],[3,4,5,6],[8,9,10,11],[12,13,14,15]] == Nothing
minimalCost [[1,1], [2,2]] == Nothing
minimalCost [[1,2], [2,2]] == Nothing
minimalCost [[1,2],[3,4]] == Nothing
minimalCost [[1]] == Nothing
minimalCost [[7, 12, 1, 5, 14], [2, 13, 8, 11], [16,3, 10, 5], [9, 6, 15, 4]] == Nothing
minimalCost [[7, 12, 1, 14], [2, 13, 8, 11], [16, 3, 10, 5], [9, 6, 15, 9, 4]] == Nothing
minimalCost [] == Nothing
minimalCost [[1,2]] == Nothing
minimalCost [[5,6]] == Nothing
minimalCost [[1,2,3]] == Nothing
minimalCost [[5,6,7]] == Nothing
minimalCost [[1,2,3], []] == Nothing
minimalCost [[5,7,6], []] == Nothing
minimalCost [[1,2,3], [5,5]] == Nothing
minimalCost [[5,7,6], [6,7]] == Nothing
minimalCost [[1,2]] == Nothing
minimalCost [[5]] == Nothing
minimalCost [[1,2]] == Nothing
minimalCost [[5,6,7]] == Nothing
minimalCost [[1,2,3]] == Nothing
minimalCost [[5,6,7,6]] == Nothing
minimalCost [[1,2,3], []] == Nothing
minimalCost [[5,6,7,6], []] == Nothing
minimalCost [[1,2,3], []] == Nothing
minimalCost [[5,7,6], [6]] == Nothing
minimalCost [[1,2,3], []] == Nothing
minimalCost [[5,7,6], [6,7]] == Nothing
minimalCost [[1,2,3], [5,5]] == Nothing
minimalCost [[5,7,6], [6,7,8]] == Nothing
minimalCost [[1,2,3], [5,5]] == Nothing
minimalCost [[5,7,6], [6,7,8]] == Nothing
minimalCost [[1,2,3], [5,5]] == Nothing
```

```
minimalCost [[5,7,6], [1,2..]] == Nothing
minimalCost [[1,2,3], [5,5]] == Nothing
minimalCost [[5,6..], [1,2..]] == Nothing
minimalCost [[1..], [5,5]] == Nothing
minimalCost [[5,6,7], [1,2..]] == Nothing
minimalCost [[1..]] == Nothing
minimalCost [ replicate 1 x | x <- [1..]] == Nothing
```