

Considereu que una llista de parells emmagatzema les arestes d'un graf no dirigit. No hi ha arestes repetides ni auto-bucles.

1. Implementeu recursivament una funció *degree* :: **Eq** *a* ⇒ [(*a*, *a*)] → *a* → **Int** que calculi el grau d'un vèrtex donat.
2. Implementeu sense recursivitat una funció *degree'* :: **Eq** *a* ⇒ [(*a*, *a*)] → *a* → **Int** que calculi el grau d'un vèrtex donat.
3. Implementeu sense recursivitat una funció *neighbors* :: **Ord** *a* ⇒ [(*a*, *a*)] → *a* → [*a*] que retorni els veïns d'un vèrtex donat en ordre creixent.

Per ordenar llistes, afegiu **import Data.List** al principi del vostre programa i utilitzeu la funció **sort** :: **Ord** *a* ⇒ [*a*] → [*a*].

Observació

A l'hora de corregir es tindrà en compte la correcció, senzillesa, elegància i eficiència de la solució proposada.

Exemple d'entrada 1

```
degree [(1, 2), (4, 5), (1, 3), (4, 1)] 1
degree [(1, 2), (4, 5), (1, 3), (4, 1)] 6
degree [('a', 'b'), ('c', 'd'), ('a', 'c'), ('x', 'a')] 'a'
```

Exemple de sortida 1

```
3
0
3
```

Exemple d'entrada 2

```
degree' [(1, 2), (4, 5), (1, 3), (4, 1)] 1
degree' [(1, 2), (4, 5), (1, 3), (4, 1)] 6
degree' [('a', 'b'), ('c', 'd'), ('a', 'c'), ('x', 'a')] 'a'
```

Exemple de sortida 2

```
3
0
3
```

Exemple d'entrada 3

```
neighbors [(1, 2), (4, 5), (1, 3), (4, 1)] 1
neighbors [(1, 2), (4, 5), (1, 3), (4, 1)] 6
neighbors [('c', 'd'), ('a', 'b'), ('a', 'c'), ('x', 'a')] 'a'
```

Exemple de sortida 3

[2, 3, 4]

[]

"bcx"

Informació del problema

Autor : Jordi Petit, Edelmira Pasarella

Generació : 2024-04-30 20:14:19

© *Jutge.org*, 2006–2024.

<https://jutge.org>