
Python — Funcions amb llistes**P51956_ca**

En aquest problema heu d'implementar una sèrie de funcions sobre llistes en Python.

1. Feu una funció `@myLength(L)@` que, donada una llista, retorni la seva llargada.
2. Feu una funció `@myMaximum(L)@` que, donada una llista no buida, retorni el seu màxim.
3. Feu una funció `@average(L)@` que, donada una llista de nombres no buida, retorni la seva mitjana.
4. Feu una funció `@buildPalindrome(L)@` que, donada una llista, retorni el palíndrom que comença amb la llista invertida.
5. Feu una funció `@remove(L1, L2)@` que donada una llista *L1* i una llista *L2*, retorna la llista *L1* havent eliminat totes les ocurrences dels elements en *L2*.
6. Feu una funció `@flatten(L)@` que aplanarà recursivament una llista d'elements que també poden ser llistes amb diferents nivells. Pista: feu-la recursiva usant `@isinstance(x, list)@`.
7. Feu una funció `@oddsNevens(L)@` que, donada una llista d'enters, retorni dues llistes, una que conté els senars i una que conté els parells, en el mateix ordre relatiu que a l'original.
8. Feu una funció `@primeDivisors(n)@` que retorni la llista de divisors primers d'un enter estrictament positiu.

Puntuació

Cada funció puntua 12 punts i l'exemple 4.

Exemple de sessió

```
>>> myLength([1, 3, 6, 1])
4
>>> myMaximum([4, 3, 1, 5, 4, 5, 2])
5
>>> myMaximum(['josep', 'jordi', 'albert'])
josep
>>> average([1, 2, 3])
2.0
>>> buildPalindrome(['pa', 'amb', 'oli'])
['oli', 'amb', 'pa', 'pa', 'amb', 'oli']
>>> flatten([[2, 6], [[8, 1, 4], [3, 'uau']], [[], [1]], [[]]])
[2, 6, 8, 1, 4, 3, 'uau', 1]
>>> remove([1, 4, 5, 3, 4, 5, 1, 2, 7, 4, 2], [2, 4])
[1, 5, 3, 5, 1, 7]
>>> oddsNevens([1, 4, 5, 3, 4, 5, 1, 2, 7, 4, 2])
([1, 5, 3, 5, 1, 7], [4, 4, 2, 4, 2])
>>> primeDivisors(255)
[3, 5, 17]
```

Informació del problema

Autoria: Jordi Petit

Generació: 2026-01-25T11:08:15.734Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>