
Clojure — Exercici de fold/reduce**Y93083_ca**

En aquest problema heu d'implementar una sèrie de funcions en Clojure utilitzant la funció `@reduce@`. No cal que pregunteu si podeu fer servir funcions auxiliars, és evident que sí.

1. Feu una funció *power* que, donats un enter x i un natural p , retorni x elevat a p , és a dir, x^p .
2. Feu una funció *my-count* que, donada una llista d'enters, calculi la seva mida.
3. Feu una funció *average* que, donada una llista d'enters no buida, calculi la seva mitjana.
4. Feu una funció *build-palindrome* que, donada una llista, retorni el palíndrom que comença amb la llista invertida.
5. Feu una funció *odds-n-evens* que, donada una llista d'enters, retorni dues llistes (en un altre llista), una que conté els parells i una que conté els senars, en el mateix ordre relatiu que a l'original.
6. Feu una funció *remove-list* que donada una llista d'enters x i una llista d'enters y , retorna la llista x havent eliminat totes les ocurrences dels elements en y .

Observacions

Per resoldre aquest problema no podeu utilitzar les funcions que ja incorpora Clojure com la del valor absolut o la potència.

Exemple d'entrada 1

```
(power 2 3)
(my-count '(2 1 3))
(average '(2 1 3))
(build-palindrome '(2 1 3))
(odds-n-evens '(1 2 3))
(remove-list '(1 2 3 4) '(1 2))
```

Exemple de sortida 1

```
8
3
2
(3 1 2 2 1 3)
((1 3) (2))
(3 4)
```

Informació del problema

Autoria: Albert Rubio / Jordi Petit / Gerard Escudero

Generació: 2026-01-25T19:57:47.797Z

© Jutge.org, 2006–2026.

<https://jutge.org>