

---

**Clojure — Seqüència Lazy****V35852\_ca**

---

Recordeu la definició circular de llistes "infinites" que vam veure a classes de laboratori?

```
(def naturals (lazy-seq (cons 0 (map inc naturals))))  
(def factorials (lazy-seq (cons 1N (map * factorials (iterate inc 1N)))))
```

Ara suposem que tenim una funció  $f$  que serveix per definir una seqüència  $a_0, a_1, a_2 \dots$  on:

$$a_0 = 1$$

$$a_{n+1} = f(a_n)$$

Així doncs, la seqüència és  $a_0, f(a_0), f(f(a_0)) \dots$ .

Definiu ara una llista "infinita", anomenem-la *seq-general*, amb els elements de la seqüència  $a_0, a_1, a_2 \dots$  fent servir aquesta tècnica de la definició circular. És a dir, cal que definiu *seq-general* tal que la seva definició tingui la forma:

```
(def seq-general ...transformació, que inclou f, de la mateixa llista seq-general)
```

**Observacions**

Afegiu el codi (*defidentity*) en el vostre codi per a que funcioni l'enviament.

**Exemple d'entrada 1**

```
(def f inc)  
(take 5 seq-general)
```

**Exemple de sortida 1**

```
#'user/f  
(1 2 3 4 5)
```

**Informació del problema**

Autoria: Jordi Delgado / Gerard Escudero

Generació: 2026-01-25T13:08:08.722Z

© Jutge.org, 2006–2026.

<https://jutge.org>