

---

**Clojure — A Base 10****U36688\_ca**

---

Feu una funció `transforma_a_base_10` que, donada una base  $b$  ( $2 \leq b \leq 9$ ), retorna una altra funció (anomenem-la `fret`): `(def fret (transforma_a_base_10 b))`. Aquesta funció `fret` retornada ha de ser tal que espera rebre un dígit  $d$  en base  $b$  ( $0 \leq d < b$ ) que forma part del nombre, en base  $b$ , que volem expressar en base 10. La crida `(fret d)` retorna una altra funció que fa el mateix procés. Tot plegat acaba en cridar aquesta funció retornada amb `-1`, aleshores es retornarà el nombre format pels dígit que hem anat passant successivament a les funcions retornades (el primer és considerat el dígit de més pes), però expressat en base 10.

Exemple: Volem passar  $101011$  (en base 2) a base 10 (on  $101011_2 = 43_{10}$ ).

Cal fer `(def h2 (transforma_a_base_10 2))`, i després només cal cridar successivament les funcions retornades per les crides a `h2` amb els dígit de  $101011$ , acabant amb `-1`: `(((((h2 1) 0) 1) 0) 1) 1) -1)` retorna 43.

**Observacions**

- No podeu fer servir cap mena d'estructura de dades, ni cap mena de seqüència. Només nombres i funcions.

**Exemple d'entrada 1**

```
(let [fret (transforma_a_base_10 2)] (((((((fret 1) 0) 1) 0) 1) 1) -1))
```

**Exemple de sortida 1**

43

**Exemple d'entrada 2**

```
(let [fret (transforma_a_base_10 3)] (((((((fret 2) 2) 1) 0) 2) -1))
```

**Exemple de sortida 2**

227

**Exemple d'entrada 3**

```
(let [fret (transforma_a_base_10 5)] (((((((((((fret 4) 3) 2) 0) 1) 4) 3) 3) -1))
```

**Exemple de sortida 3**

365868

**Exemple d'entrada 4**

```
(let [fret (transforma_a_base_10 9)] (((((((((((fret 8) 7) 1) 0) 3) 5) -1))
```

## **Exemple de sortida 4**

519080

## **Informació del problema**

Autoria: Jordi Delgado

Generació: 2026-01-25T12:59:54.970Z

© *Jutge.org*, 2006–2026.

<https://jutge.org>