



## Problema 7

18 / 21 Novembre 2019

### Descrizione

In questi esercizi ti viene richiesto di lavorare con sequenze ordinate di numeri interi, rappresentate attraverso le liste. Per operare sulle liste utilizza esclusivamente la costante `null` e le primitive di base: `null?`, `car`, `cdr`, `cons`.

Innanzitutto, definisci in Scheme una procedura `belong?` che, dati un intero  $x$  e una lista ordinata  $S$ , verifica se  $x$  è un elemento di  $S$ .

### Esempi

```
(belong? 18 '()) → false
(belong? 18 '(5 7 10 18 23)) → true
(belong? 18 '(5 7 10 12 23)) → false
```

Successivamente, definisci in Scheme una procedura `position` che, dati un intero  $x$  e una lista ordinata e senza ripetizioni  $S$ , restituisce la posizione (indice) di  $x$  in  $S$ .

### Esempi

```
(position 7 '(7 8 24 35 41)) → 0
(position 35 '(7 8 24 35 41)) → 3
(position 41 '(7 8 24 35 41)) → 4
```

Definisci quindi una procedura `sorted-ins` che, dati un intero  $x$  e una lista ordinata e senza ripetizioni  $S$ , restituisce la lista *ordinata e senza ripetizioni* che contiene  $x$  e tutti gli elementi di  $S$ .

### Esempi

```
(sorted-ins 24 '()) → '(24)
(sorted-ins 5 '(7 8 24 35 41)) → '(5 7 8 24 35 41)
(sorted-ins 24 '(7 8 24 35 41)) → '(7 8 24 35 41)
(sorted-ins 27 '(7 8 24 35 41)) → '(7 8 24 27 35 41)
```

Infine, applica `sorted-ins` per definire una procedura `sorted-list` che, data una lista senza ripetizioni  $S$ , restituisce la lista *ordinata* e senza ripetizioni  $S'$  che contiene tutti e soli gli elementi di  $S$ .

### Esempio

```
(sorted-list '(35 8 41 24 7)) → '(7 8 24 35 41)
```