Advanced Use of Generics.

```
(define ((Lagrange-equations Lagrangian) config-space-path)
  (let ((state-path (Gamma config-space-path)))
    (- (D (compose ((partial 2) Lagrangian) state-path))
       (compose ((partial 1) Lagrangian) state-path))))

(define ((Gamma w) t)
  (up t (w t) ((D w) t)))

(define ((L-harmonic m k) state)
  (- (* 1/2 m (square (velocity state)))
     (* 1/2 k (square (coordinate state)))))

(((Lagrange-equations (L-harmonic 'm 'k))
  (literal-function 'x))
 't)
#| (+ (* k (x t)) (* m (((expt D 2) x) t))) |#
```