

Προγραμματιστικές Τεχνικές (2016 - 2017)

Programming Task 2

Return to:
Assignments
and... ➔

Version 1 -- 13/10/2016 23:52

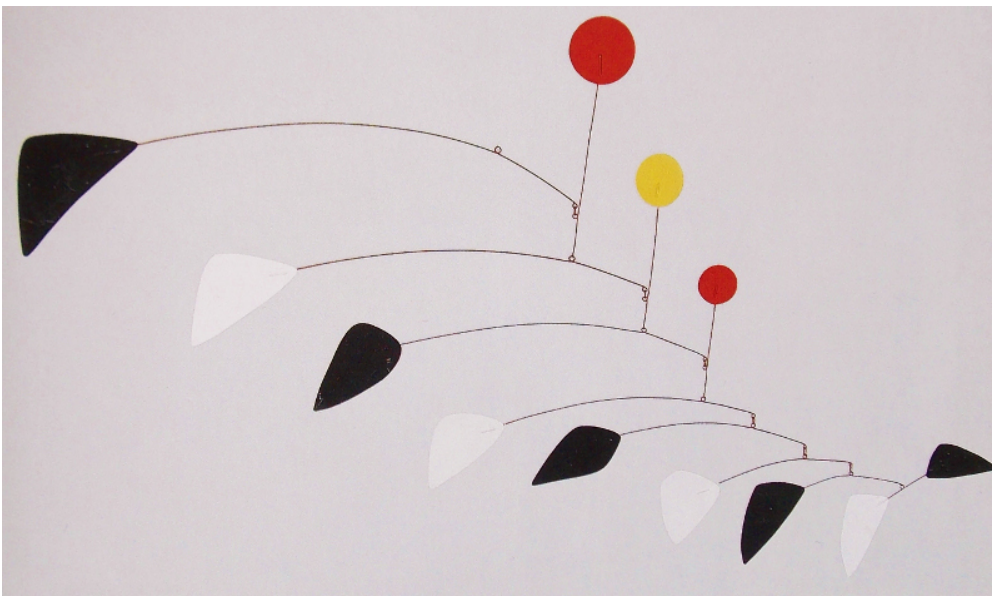
Κατάθεση εργασίας για βαθμολογία

1. Κατεβάζετε τα 2 συνοδευτικά αρχεία της Άσκησης, με τον κώδικα (ps2.scm) και τα τεστ (tester2.scm).
2. Συμπληρώνετε τα τμήματα του κώδικα που λείπουν στο πρώτο.
3. Συμπληρώνετε επιπλέον τεστ στο δεύτερο --παρόμοιο αρχείο θα εκτελεστεί στον server για να βαθμολογηθείτε.
4. Ανεβάστε το πρώτο αρχείο με τις δικές αλλαγές στο CMS
5. Επαναλάβετε τα βήματα 2-4 μέχρι η βαθμολογία να σας είναι ικανοποιητική, ή παρέλθει ο χρόνος παράδοσης, ή βρείτε να κάνετε κάτι καλύτερο με το χρόνο σας.

Alexander Calder

Η Wikipedia περιγράφει ένα mobile ως εξής:

Ένα mobile είναι μία κατασκευή που εκμεταλλεύεται τις αρχές της ισορροπίας. Αποτελείται από έναν αριθμό ράβδων (bar), από τις οποίες κρέμονται γλυπτά (sculpture) ή άλλες ράβδοι. Τα αντικείμενα ισορροπούν μεταξύ τους, με αποτέλεσμα οι ράβδοι να παραμένουν σχεδόν οριζόντιοι και όλο το σύστημα να ισορροπεί. Κάθε ράβδος κρέμμεται από ένα μόνο νήμα (string), δίνοντάς της με αυτό τον τρόπο τη δυνατότητα να περιστρέφεται γύρω από αυτό το νήμα.



Three Disks in the Air mobile του Alexander Calder (1967)

Θα χρησιμοποιήσουμε την εξής αυστηρή περιγραφή. Ένα mobile μπορεί να είναι:

- είτε γλυπτό
- είτε ράβδος που κρατάει δύο mobile.

Στην πρώτη περίπτωση, αναπαριστά ένα ατομικό γλυπτό και χαρακτηρίζεται από έναν αριθμό που το βάρος του σε `kg`. Στην δεύτερη περίπτωση, αναπαριστά τα δύο mobile και το βάρος της ράβδου.

Θεωρώντας δεδομένες τις συναρτήσεις:

- `(make-mobile left weight right)`, η οποία δημιουργεί μία ράβδο με βάρος *weight* και αριστερά και δεξιά κλαδιά *left* και *right* αντίστοιχα,
- `(left-mobile m)`, η οποία επιστρέφει το αριστερό μέρος του mobile *m*,
- `(right-mobile m)`, η οποία επιστρέφει το δεξί μέρος του mobile *m*,
- `(is-sculpture? m)`, η οποία επιστρέφει `#t` (*True*) αν είναι γλυπτό, αλλιώς `#f` (*False*) και
- `(get-weight m)`, η οποία επιστρέφει το βάρος ενός mobile,

Παραδείγματα χρήσης

- `(is-sculpture? (make-mobile 2))` επιστρέφει `#t`
- `(is-sculpture? (make-mobile 1 2 3))` επιστρέφει `#f`
- `(get-weight (make-mobile 2))` επιστρέφει `2`

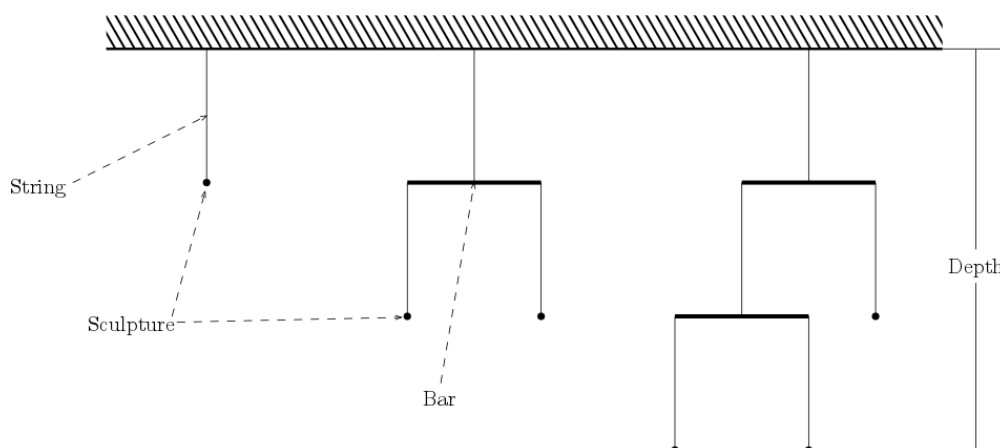
- ```
(get-weight (right-mobile (make-mobile 1
 4
 (make-mobile 5
 6
 7)))))
```

επιστρέφει 6

υλοποιήστε τις παρακάτω συναρτήσεις:

1. `(define (weights# m) '<YOUR-CODE-HERE>)`, η οποία παίρνει σαν όρισμα ένα mobile  $m$  και απαριθμεί τα βάρη του.
2. `(define (total-weight m) '<YOUR-CODE-HERE>)`, η οποία παίρνει σαν όρισμα ένα mobile  $m$  και υπολογίζει το συνολικό του βάρος. Στο βάρος ενός mobile συμπεριλαμβάνεται το βάρος των γλυπτών και το βάρος των ράβδων.
3. `(define (depth m) '<YOUR-CODE-HERE>)`, η οποία παίρνει σαν όρισμα ένα mobile  $m$  και προσδιορίζει τον μέγιστο αριθμό νημάτων για φτάσουμε σ' ένα οποιοδήποτε γλυπτό του.
4. `(define (balanced? m) '<YOUR-CODE-HERE>)`, η οποία παίρνει σαν όρισμα ένα mobile  $m$  και προσδιορίζει αν είναι ισορροπημένο. Ένα mobile είναι ισορροπημένο αν το βάρος του mobile στο ένα άκρο είναι ίσο με το βάρος του mobile στο άλλο και κάθε mobile είναι επίσης ισορροπημένο.

## Παράδειγμα τριών διαφορετικών mobile - επεξήγηση ορισμών



◀ Programming  
Task 1 Solutions

Programming  
Task 2 Solutions



[Return to: Assignments and... ➔](#)