

Assignment 10: Documentation

Due: 4:00 p.m., Friday, 12 October 2007

No extensions!

Summary: In this assignment, you will document a variety of procedures.

Purposes: To give you practice writing documentation using the Six-P format. To give you practice reading procedures, both recursive and non-recursive.

Expected Time: One to two hours

Collaboration: I encourage you to work in groups of size three. You may, however, work alone or work in a group of size two or size four. You may discuss this assignment with anyone, provided you credit such discussions when you submit the assignment.

Submitting: Email me your answer. More details below.

Warning: So that this assignment is a learning experience for everyone, I may spend class time publicly critiquing your work.

Contents:

- Assignment
- Important Evaluation Criteria
- Submitting Your Homework

Assignment

Document the following procedures using the Six-P documentation rubric. In writing your documentation, pay particular attention to the cases in which the procedure returns a result, but perhaps a different result than the documentation would normally suggest. For example, if we have a procedure that increments the red component by 32, that procedure will not increment the red component by 32 when the component is more than 223.

When one of the following procedures uses another procedure that is not defined here, you can assume that procedure behaves as its name suggests. For example, `rgb.darker` computes the darker of two colors and `rgb.reddish?` determines whether a color is reddish.

When a procedure is accompanied by a helper whose name includes the word “helper”, you should not document the helper.

```
(define rgb.grey
  (lambda (n)
    (rgb.new (* 8 n) (* 8 n) (* 8 n))))
```

```

(define average
  (lambda (v1 v2)
    (/ (+ v1 v2) 2)))

(define rgb.average
  (lambda (c1 c2)
    (rgb.new (average (rgb.red c1) (rgb.red c2))
              (average (rgb.green c1) (rgb.green c2))
              (average (rgb.blue c1) (rgb.blue c2)))))

(define rgb.enhance
  (lambda (color boundary)
    (rgb.new (if (< boundary (rgb.red color))
                  (+ (rgb.red color) 16)
                  (- (rgb.red color) 16))
              (if (< boundary (rgb.green color))
                  (+ (rgb.green color) 16)
                  (- (rgb.green color) 16))
              (if (< boundary (rgb.blue color))
                  (+ (rgb.blue color) 16)
                  (- (rgb.blue color) 16)))))

(define spots.bounds
  (lambda (spots)
    (spots.bounds-helper (spot.col (car spots))
                          (spot.row (car spots))
                          (spot.col (car spots))
                          (spot.row (car spots))
                          spots)))

(define spots.bounds-helper
  (lambda (left top right bottom remaining)
    (if (null? remaining)
        (list left top right bottom)
        (spots.bounds-helper (min left (spot.col (car remaining)))
                              (min top (spot.row (car remaining)))
                              (max right (spot.col (car remaining)))
                              (max bottom (spot.row (car remaining)))
                              (cdr remaining)))))

(define rgb-list.darkest
  (lambda (colors)
    (if (null? (cdr colors))
        (car colors)
        (rgb-list.darkest (cons (rgb.darker (car colors) (cadr colors))
                                (cddr colors))))))

(define rgb-list.all-reddish?
  (lambda (colors)
    (or (null? colors)
        (and (rgb.reddish? (car colors))
              (rgb-list.all-reddish? (cdr colors))))))

```

Important Evaluation Criteria

I will evaluate your responses primarily on their correctness, their clarity, and the care with which you specify preconditions and postconditions.

Submitting Your Homework

Please submit this work via email. The email should be titled CSC151.02 Assignment 10 and should contain your answers to all parts of this assignment.

Please send your work as the body of an email message. I don't like attachments, and prefer not to receive them when they can be avoided.

Copyright © 2007 Janet Davis, Matthew Kluber, and Samuel A. Rebelsky. (Selected materials copyright by John David Stone and Henry Walker and used by permission.) This material is based upon work partially supported by the National Science Foundation under Grant No. CCLI-0633090. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This work is licensed under a Creative Commons Attribution-NonCommercial 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/2.5/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.