

```
;;
;; CPSC 110, Fall 2010
;; Practice Problems for Midterm 1
;;
;; These problems are intended to provide you with practice exercises for
;; the first midterm. Additional practice problems covering material from
;; week 4 will be distributed at the end of that week.
;;
;; Regular lab sections during week 4 will be devoted to working through
;; these practice problems. TAs will be able to give you advice on working
;; the problems. TAs in the DLC will also be able to help with these
;; practice problems.
;;
;; NOTE: There are FAR MORE PROBLEMS HERE THAN WILL APPEAR ON AN ACTUAL
;; MIDTERM. We are giving you extra problems to help you practice.
;;
```

```
;; -----
;; Section 1
;;
;;
```

```
;; What is the value of:
(+ 2 (* 3 5))
```

```
;; What is the value of:
(string-append "Roberto" " " "Luongo")
```

```
;; What is the value of:
(< 3 4)
```

```
;; What is the value of:
(if (< 1 3)
    "dog"
    "cat")
```

```
;; Given
(define SPEED 10)
```

```
;; What is the value of

(* SPEED 2)
```

```
;; Given
(define (prod x y)
  (* x y))
```

```
;; what is the value of

(prod 3 7)
```

```
;; Given
```

```
(define (choose a b)
  (if (< a b)
      a
      b))
```

```
;; what is the value of
```

```
(choose 5 3)
```

```
;; Given
```

```
(define-struct pkg (a b c))
```

```
(define P1 (make-pkg 3 4 5))
```

```
;; what is the value of
```

```
(pkg-b P1)
```

```
;; -----
```

```
;; Section 2
```

```
;;
```

```
;; Consider this define-struct
```

```
(define-struct dims (width height length))
```

```
;; What is the name of the constructor it defines?
```

```
;; What is the name of the predicate it defines?
```

```
;; What are the names of all the selectors it defines?
```

```
;; -----
```

```
;; Problem 3
```

```
;;
```

```
;; Kinds of Data definitions
```

```
;;
```

```
;; In this problem you will be given several small fragments of
;; a problem description. Each describes some information in a
;; problem domain that must be represented using data in a program.
```

```
;;
```

```
;; In each case say whether the data definition you would use is:
```

```
;;   unrestricted atomic    (Integer, String, Number etc.)
```

```
;;   interval
```

```
;;   enumeration
```

```
;;   itemization of intervals
```

```
;;   compound
```

```
;;
```

```
;; CHOOSE ONLY ONE, THE MOST SPECIFIC
```

```
;;
```

```
;; You may find it helpful in each case to assume you are working
```

```
;; on a typical world program, in which typical constants like WIDTH,
```

```

;; HEIGHT, etc. have been defined.
;;
;; You may make any other reasonable assumptions you wish, but if they
;; are essential to the correctness of your answer write your assumption
;; down.
;;

;;
;; the x coordinate of the cat
;;

;;
;; the height of the rocket
;;

;;
;; the current point in a countdown from 10 to 0
;;

;;
;; whether the flashing light is yellow or dark
;;

;;
;; the x and y coordinate of a plane
;;

;;
;; the color and height of a sparkling firework
;; (where sparkling means that its color keeps changing)
;;

;; -----
;; Problem 4
;;
;; Designing Data Definitions
;;

;; In this problem you will design complete data definitions for several different
;; kinds of program domain information that must be represented as data in
;; a program. Assume you have access to the HtDDD web page. You should
;; consider all of these to be SEPARATE PROGRAMS.
;;
;; For some of these questions you may find it helpful in each case to
;; assume you are working on a typical world program, in which typical
;; constants like WIDTH, HEIGHT, etc. have been defined.
;;
;; You may make any other reasonable assumptions you wish, but if they
;; are essential to the correctness of your answer write you assumption
;; down.

;; the x coordinate of a flying saucer

```

```
;; the ice cream flavor, one of chocolate, vanilla or strawberry
```

```
;; a score, which is an integer ranging from 0 to 5 inclusive
```

```
;; a reading, which an integer in one of the ranges [0, 3) [3, 7) [7, 9]
```

```
;; a person, with a first name and a last name
```

```
;; a plane, with a altitude, compass direction and speed  
;; Note: compass direction means an integer in [0, 359)
```

```
;; -----  
;; Problem 5
```

```
;; Given this data definition:
```

```
;; Weight is Number  
;; interp. a person's weight in kilos  
(define W1 52)  
(define W2 72)  
#;  
(define (fn-for-w w)  
  (... w))
```

```
;; Design a function called half-weight that consumes a  
;; Weight and produces another Weight that is half as much.  
;; Please explicitly show your stub, as well as your template.
```

```
;; ----  
;; Given this data definition:
```

```
;; Name is String  
;; interp. a dog's name  
(define N1 "spot")
```

```

(define N2 "rover")
#;
(define (fn-for-name n)
  (... n))

;; Design a function called long-name? that consumes a Name and produces
;; true if the length of the name is longer than 8 and false otherwise.
;; Please explicitly show your stub, as well as your template.
;;
;; You may want to use the BSL string-length function, which has the signature
;; String -> Natural, and produces the length of the string it is provided.


;; ----
;; Given this data definition:

(define-struct fw (y fuse))
;; Firework is (make-fw Integer Natural)
;; interp. a firework, y is height, fuse is number of ticks before it explodes
(define FW1 (make-fw 400 99))
(define FW2 (make-fw 47 1)) ;this one is about to explode

(define (fn-for-fw f)
  (... (fw-y f) (fw-fuse f)))

;; Design a function named exploded? that consumes a Firework and produces true if
the
;; its fuse is 0 and false otherwise.
;; Please explicitly show your stub, as well as your template.

;; Given the same Firework data definition, design a function named tick-fw that
consumes
;; a Firework and produces a new Firework as follows:
;;   the new firework's y should be the old y minus SPEED.
;;   the new firework's fuse should be 1 less than the old fuse,
;;   except it should never be less than 0
;; the fuse should countdown to 0 (and never go below 0)
;;
;; Please explicitly show your stub, as well as your template.


;; -----
;; Problem 6
;;
;; Given this partial program:
;;

;; A Bouncing Ball

(require 2htdp/image)
(require 2htdp/universe)

```

```

;;
;; Constants:

(define WIDTH 200)
(define HEIGHT 400)

(define X-POS (/ WIDTH 2))

(define BALL (circle 10 "solid" "blue"))

(define MTS (empty-scene WIDTH HEIGHT))

;; Data definitions:

;; Location is Number
;; interp. the y coordinate of a BALL
(define L1 HEIGHT)
(define L2 (/ HEIGHT 2))
(define L3 0)
#;
(define (fn-for-location l)
  (... l))

;; Functions

;; Design a function named render-ball that consumes a Location and
;; produces an image in which BALL is in the appropriate place on MTS.
;;
;; You may want to use the function place-image, which has the signature
;; Image Number Number Image -> Image. It places its first argument
;; onto its second argument at the x, y position specified by its
;; second and third arguments.

```