CONS CAR CDR LIST.

cons and list are used to create data structures. cons creates a “cons cell”—a pair of two items. list creates a chain of cons cells terminated in a nil, also known as the empty-list object '().

Draw the box-and-pointer diagram created after the corresponding expression is evaluated.

Problem 1.

(define a (cons 1 2))

Problem 2.

(define b (cons 1 (cons 2 3)))

Problem 3.

(define nil '())
(define c (cons 1 (cons 2 nil)))
Problem 4.

(define d (list 1 2 3))

Problem 5.

(define e (list (cons 1 2) (list 3 4)))
car and cdr are used to unpack cons cells. car extracts the first thing in the cons cell, and cdr gets the second thing.

Remember that a list is a chain of cons cells.

For the following problems, write the value that is produced by evaluating the expression. If the thing produced is a list, make sure to write its open and close parentheses.

Problem 6.

(car (cons 1 2))

answer: 1

Problem 7.

(cdr (cons 1 2))

Problem 8.

(car (list 1 2 3))

Problem 9.

(cdr (list 1 2 3))

Problem 10.

(car (cdr (list 1 2 3)))
There are shortcut expressions for a series of car and cdr operations. E.g., cadr takes the car of the cdr of an object. Note that this is the second item of a list.

All permutations of “c…r” exist with up to four a’s and d’s in between. E.g., cadddr is the car of the cdr of the cdr of the cdr of a list.

Given this list-of-lists:

```
(define foo (list 1 2 (3 4)))
```

write expressions to yield the following objects. You may use a series of cars andcdrs or the c...rc shortcuts:

**Problem 11.**

1

answer: (car foo)

**Problem 12.**

2

**Problem 13.**

(3 4)

**Problem 14.**

3

**Problem 15.**

4