

## Programming in Lisp

### Exam #1 Review

This worksheet is designed to allow students to work through problems similar to those that will be found on the exam. A student who feels comfortable completing these problems is well prepared for the exam.

The exam will cover the following topics:

- Differences between Lisp and other languages
- Atoms & Lists
- Expressions
- Evaluation
- List operators (list, listp, copy-list, append, cons, car cdr, nth, nthcdr, first, second, ..., last)
- Dotted lists
- Predicates (null, not, >, <, >=, <=, /=)
- If, and, or
- Format
- Progn
- Let and local variables
- Mapping functions (mapcar)
- Lambda functions
- Sequence Operators (member, length, reverse, sort)
- Arrays
- Structures
- Do
- Cond
- Function defining and parameters (&key, &rest, &optional)
- Closures

The exam covers chapters 2-7 in Graham.

Programming in Lisp  
Exam #1 Review

1. Given the following definition of a triangle, write a function to take a list of lengths and return a list of equilateral triangles whose sides are the lengths in the list.

```
(defstruct triangle a b c)
```

2. Write a function to produce n rows of stars where row  $i = 1..n$  has  $i$  stars:

```
> (stars 5)
*
**
***
****
*****
```

Programming in Lisp  
Exam #1 Review

3. Write a function that takes three arguments: `len`, `oldarray`, and `addit`. `Addit` should be optional, and default to 0. The function should return a new array, containing the contents of `old-array` + `addit`, for indexes between 0 and `len-1`. That is:  $\text{NewArray}[i] = \text{OldArray}[i] + \text{Addit}$ , for  $0 \leq i < \text{len}$ .

4. An alternative to `(defstruct rectangle length width)` is to use a *cons* to hold the length and width.

a. Write a function **make-rectangle** that takes two keyword arguments, `:length` and `:width` (which both default to 0) and creates such a cons.

b. Write a function **rectangle-width** which returns the width given such a cons. Repeat for length.

Programming in Lisp  
Exam #1 Review

5a. Write a three field structure to hold the radius, area and circumference of the circle. If given only the radius, the area and circumference should default to the correct values.

b. Write a function that given two radii, if the first is larger, the area of the corresponding circle is returned (use the structure from part a to compute the area). If the second radii is larger, the function should return the corresponding circumference. If the radii are equal, return 'EQUAL. Use *cond* as appropriate.

c. Write a function that takes any number of radii as arguments, and returns the largest area. Use the function *sort* as appropriate.

Programming in Lisp  
Exam #1 Review

d. Write a function that takes a circle (check it!) and a height, computes the volume of the corresponding cylinder ( $V = \pi r^2 h$ ), the surface area ( $A = 2\pi r h$ ), determines which is greater and prints all of this information (volume, surface area, which is greater) without computing a value twice, i.e. use local variables.

e. Write a closure which takes as input a circle and returns the largest area of a circle ever passed to it. It should return nil if no circles have been passed.