

CIS-280: Lab-5
Working with Sequences
Monday, March 14, 2005

1. Define `x`, `y`, `z`, and `w` as follows:

```
(define x '(5 4 3))
(define y (list 66 67 68))
(define z '(35 cat dog 764))
(define w (list 67 'paper 'x x z))
```

Now examine the structure of each object by typing the object's name at the prompt. Make sure you understand the reason for what appears.

2. Now do the following operations, observe what structure is returned, and make certain you understand why.

```
(car x)
(cdr x)
(cdr z)
(car (cdr z))
(cons 98 x)
(cons 'apple x)
(cons 7 (cons 'duke x))
(cons x y)
(append x y)
```

3. By using `cons`, `list`, and `append`, define the named objects as the following structures: (Use only one statement for each definition and try to use the structures defined earlier in the problem in your definition. For example, the list structure for `v` can be defined as `(define v (append x y))`)

| Object Name | Structure |
|----------------|--|
| <code>x</code> | <code>(5 4 3)</code> |
| <code>y</code> | <code>(66 67 68)</code> |
| <code>u</code> | <code>(boy girl)</code> |
| <code>v</code> | <code>(5 4 3 66 67 68)</code> |
| <code>a</code> | <code>(125 5 4 3 66 67 68)</code> |
| <code>b</code> | <code>(cat 5 4 3 66 67 68)</code> |
| <code>c</code> | <code>(rattle 987 cat 5 4 3 66 67 68)</code> |
| <code>d</code> | <code>((5 4 3) (66 67 68))</code> |
| <code>e</code> | <code>((boy girl) (5 4 3) (66 67 68))</code> |
| <code>f</code> | <code>((5 4 3) (66 67 68 boy girl))</code> |

To see if you defined the structure correctly, merely type the object name (such as `v`) at the prompt, and see if the correct list structure appears.

4. Write a procedure (`AllLarger elt lst`) that takes as arguments an integer `elt` and a list of integers `lst`, and returns true if all the elements in `lst` are larger than `elt` and false otherwise.
5. Write a procedure (`RepeatElt lst`) that takes as argument a list of integers `lst`, and returns the list with each element repeated twice. For example,

```
(RepeatElt '(5 3 1 7 1 4 8)) returns (5 5 3 3 1 1 7 7 1 1 4 4 8 8)
```

Submit your definitions from problem 3 and your code for `AllLarger` and for `RepeatElt` on the electronic submission system for Lab-5.