

# Style Sheet for Lisp

Version of 020724

1. Unless otherwise needed, indent each level by two spaces. Do not indent using the TAB key. The notation “IND” means that your indentation is incorrect.
2. Vertically align items of a list when they appear on separate lines: For example, if  $\alpha$  and  $\beta$  are long expressions, write

```
(list  $\alpha$ 
       $\beta$ )
```

or

```
(list  $\alpha$ 
       $\beta$ 
)
```

If  $\alpha$  and  $\beta$  are short expressions, then `(list  $\alpha$   $\beta$ )` may be more appropriate.

3. If you create helper functions, document them and give them appropriate names.
4. Documenting non-trivial functions should include a hand-trace which shows (1) the results expected for a representative set of input; and (2) the function calls by which those results are computed.
5. Lisp keywords should be in lowercase. Identifiers which *you* create should be all uppercase. Use capitalization consistently.
6. Do not use global variables unless they are truly required.
7. Do not put a `let` directly outside a `do`. It's never required.
8. Use `cond` instead of nested `ifs`.
9. Avoid nested `conds` and nested `ifs`.
10. Names of functions and variables should reflect the purpose of said functions and variables. The notation `BFN` means “Bad Function Name” and the notation `BVN` means “Bad Variable Name.”
11. Include your name in a comment near the very top of each code file. The grading notation “`NNIC`” means “No Name In Code.”
12. Hardcopy of Lisp code, data, and output should be printed using a fixed-width font. “`UFWF`” is the notation for “Use Fixed Width Font.”
13. When printing, be sure that single lines of code are not split across several output lines. Avoid wraparound (“`WA`”).
14. Don't apply `car` or `cdr` to `nil`.

15. When I give you test cases to try, put these in the hardcopy *before* any other test cases you may try.

## COMMON ERRORS

1. Don't return 'nil if nil is sufficient. Same for '2 or '3.7.
2. Don't use forms like `(cons  $\alpha$  (cons  $\beta$  nil))`. Use `(list  $\alpha$   $\beta$ )` instead.
3. Don't use `(eq  $\alpha$  t)` when  $\alpha$  alone will work.  
Ditto for `(eq  $\alpha$  nil)` and `(not  $\alpha$ )`.  
Thus, write `(cond ( $\alpha$  ...) ...)`  
instead of `(cond ((eq  $\alpha$  t) ...) ...)`.