

CS 3723, Example showing: Opening a file, reading file, evaluating

```
runner% cat assign1.data
(setq l '(a b))
(list l l l)
(append l l l)
(cons l l)
(setq x '(a b))
(cons (car x) (cons (cadr x) '(c d)))
end

runner% cat open-read-eval.1
(defun open-read-eval (filename)
  (setq file (open filename :direction :input))
  (loop
    (let ((in (read file)))
      (if (equal in 'end) (return nil))
      (terpri)
      (princ "Input> ")(prin1 in)
      (terpri)
      (princ "Value: ")(prin1 (eval in))
      (terpri) )))

runner% lisp
> (load "open-read-eval.1")
;;; Loading source file "open-read-eval.1"
;;; Warning: File "open-read-eval.1" does not begin with IN-PACKAGE. Loading into
package "USER"
#P"/home/faculty/wagner/courses/CS3323/open-read-eval.1"
> (open-read-eval "assign1.data")

Input> (SETQ L (QUOTE (A B)))
Value: (A B)

Input> (LIST L L L)
Value: ((A B) (A B) (A B))

Input> (APPEND L L L)
Value: (A B A B A B)

Input> (CONS L L)
Value: ((A B) A B)

Input> (SETQ X (QUOTE (A B)))
Value: (A B)

Input> (CONS (CAR X) (CONS (CADR X) (QUOTE (C D))))
Value: (A B C D)
NIL
> (quit)
runner%
```