FCPL – Test 1 – Tatu Bogdan

1. (rotate-right (rotate-right '(a (b c) ())) Implement and give the result!

```
(defun rotate_right (lst)
   (setq reversed_lst (reverse lst))
   (cons (car reversed_lst) (reverse (cdr reversed_lst)))
)
```

Result: ((B C) NIL A)
* NIL or (), it is the same thing

2. member Explain the functioning. Give two examples!
Member takes 2 arguments and tries to find the 1st arg in the 2nd (list)
If it finds the member it returns the list from that member onward, else it returns nil

```
(member 'b '(a b c d))
# (b c d)
(member 'f '(a b c d))
# NIL
```

3. Nil , Null and () ! Compare it ! Examples!
Null is a predicate and it takes an argument
It returns true if argument is NIL or ()
NIL and () are the same

4.

```
(SETQ B 'C) ; sets B as C
# C
(SETQ C 'D) ; sets C as D
# D
(EVAL B ) ; evals B -> C which is D
# D
```

EVAL determines another evaluation of the argument
B would print C, so (EVAL B) evaluates B as C, then C would print D
Give the result and explain !

5. (CONS '(B C) '(A B C)) Explain the result and give it!
((B C) A B C)
Sets (B C) as head of list (A B C) so it returns ((B C) A B C)