Name : Pritesh Ratnappagol

1. Length of the list:

* The program can be used to compute the length of the given list of various sizes.
* Program also checks if there are no elements in the list then returns “ 0 ” (i.e. zero) else calculates the length of the list accordingly and returns the value.
* To implement this I am first checking if list contains elements in it or not; If no elements it will return zero else it will calculate the length of the list by recursively performing addition operation on the tail of the list.
* (defun len (list)

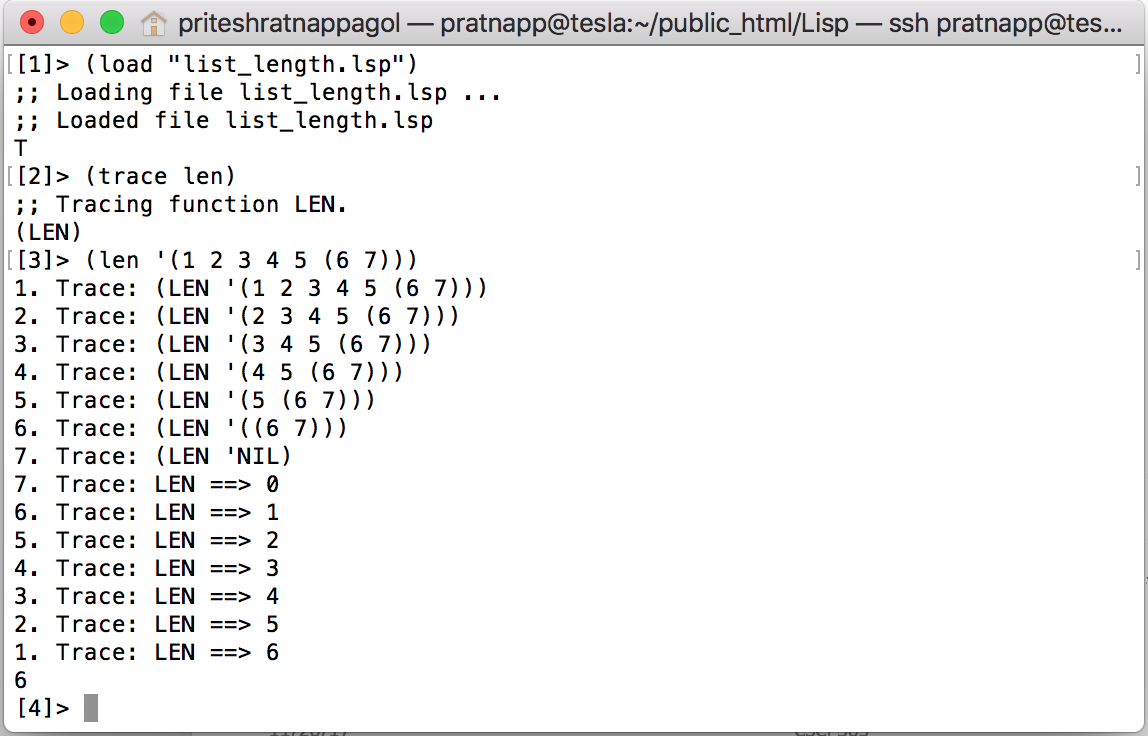
(if list

(1+ (len (cdr list)))

0))



Screenshot 1. Instances of finding length of the list

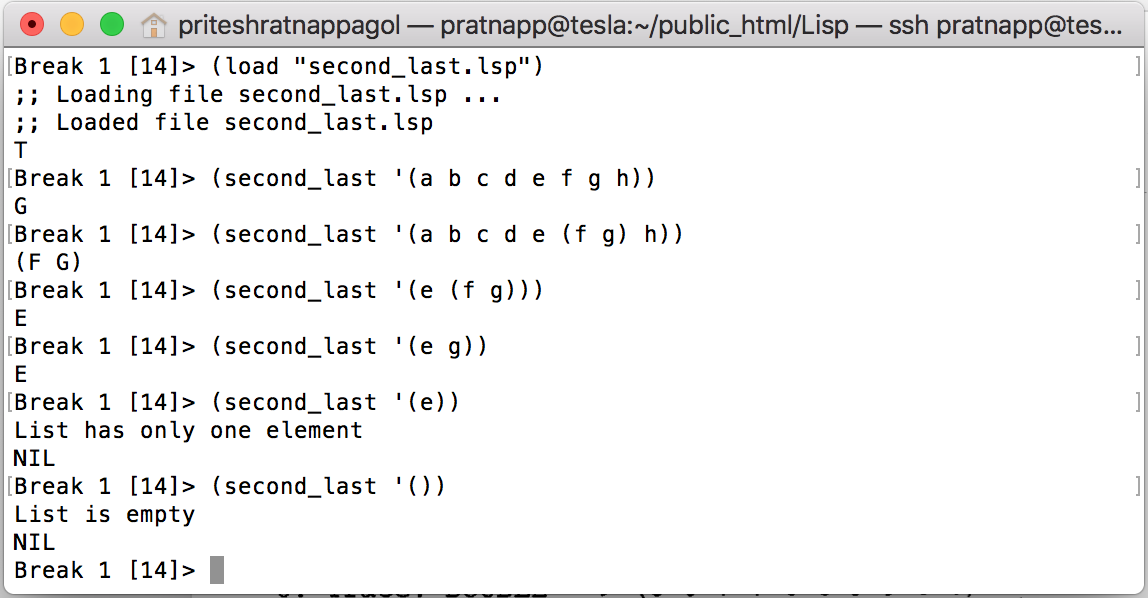


Screenshot 2. Trace of finding length of the list

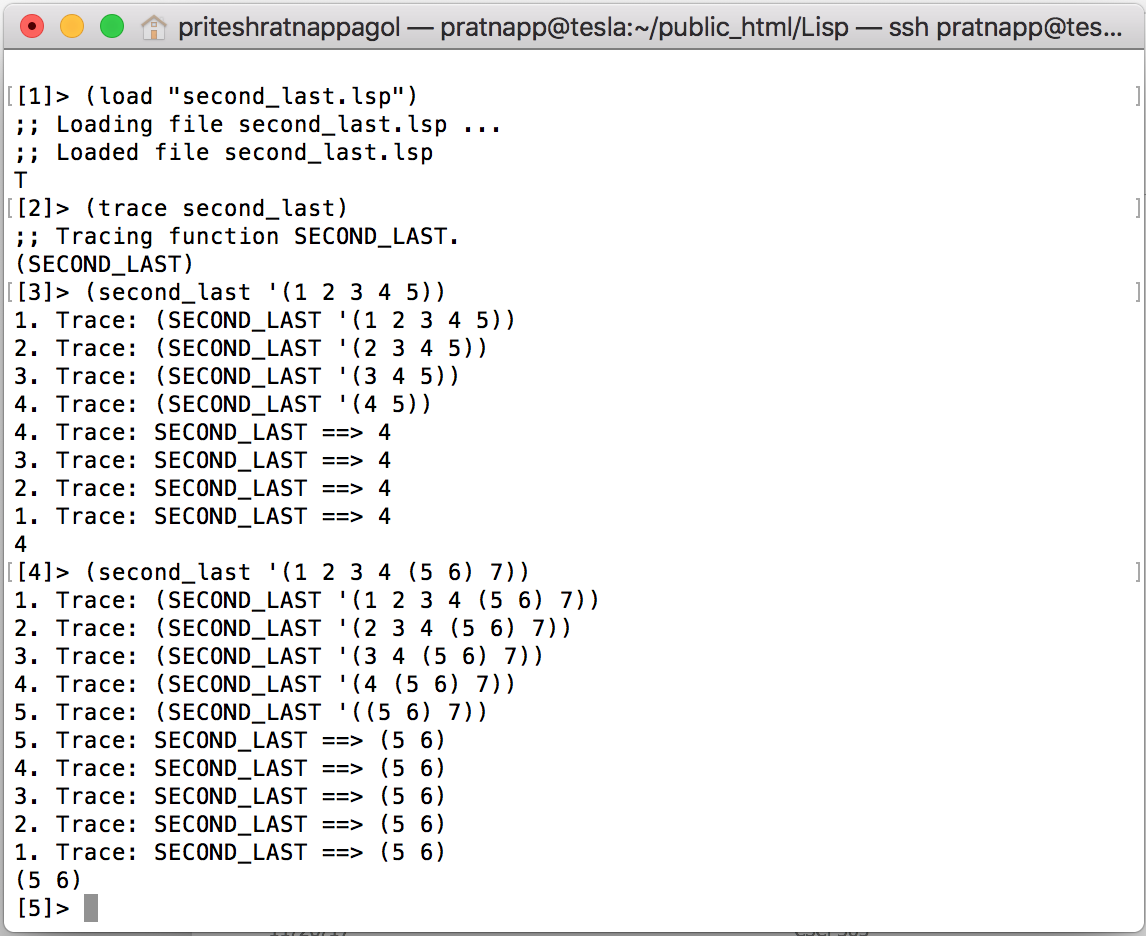
1. Second to last element of the given list:

I have implemented it using four conditions:

1. Program first checks if list is empty and if so it returns the statement “List is empty” and exits the program
2. If list is not empty and contains only one atom in it, it returns “List has only one item” and exits the program
3. If list is not empty and contains two element, it returns the second last element by simply using the car function for the given list.
4. If list is not empty and contains more than two elements it recursively performs the above steps on tail of the list using cdr function till it is reached the step 3 and returns the second last element in the list.



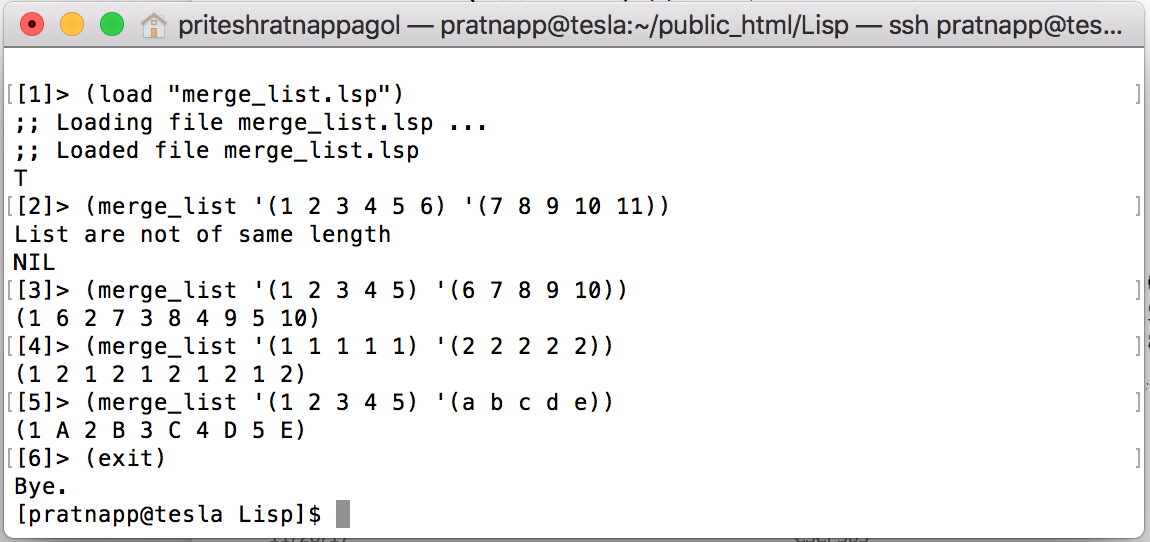
Screenshot 3. Instances of finding second to last element of the given list



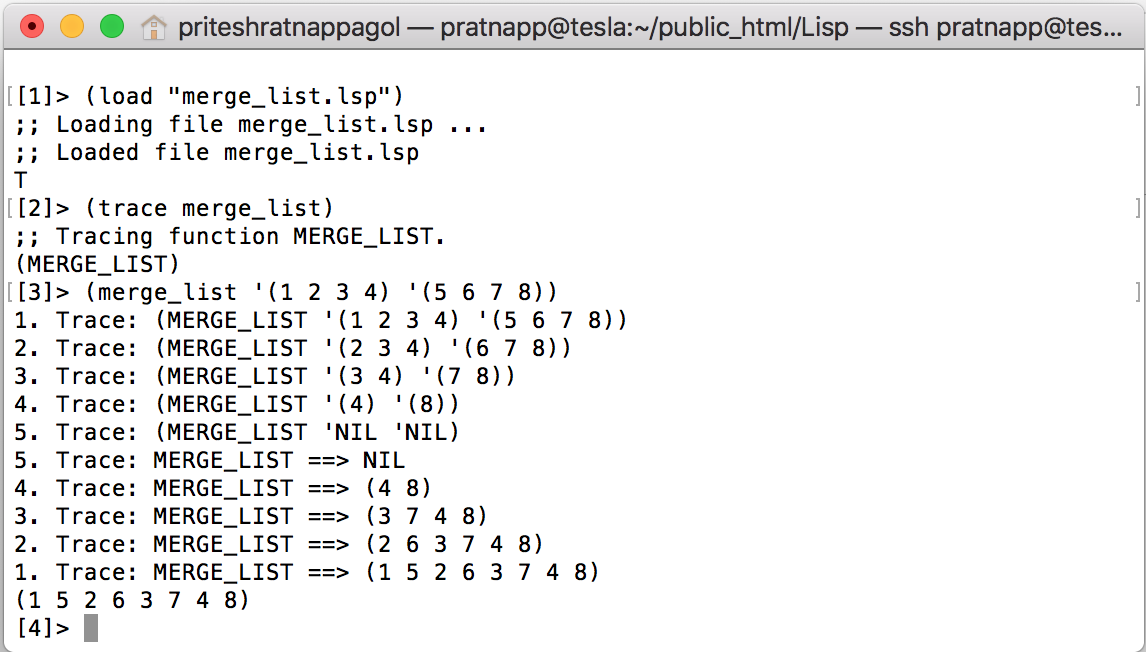
Screenshot 4. Trace of finding second to last element of the given list

1. Perfect shuffle operation on first and second list:

To implement this function I am taking two lists of equal length from the user and checking the condition if length of the two list is equal or not and creating a new list by using the car function on both the lists for getting the first element from both the lists and recursively calling the con function on the tail of both the lists using cdr.



Screenshot 5. Instance on shuffle operation on two lists

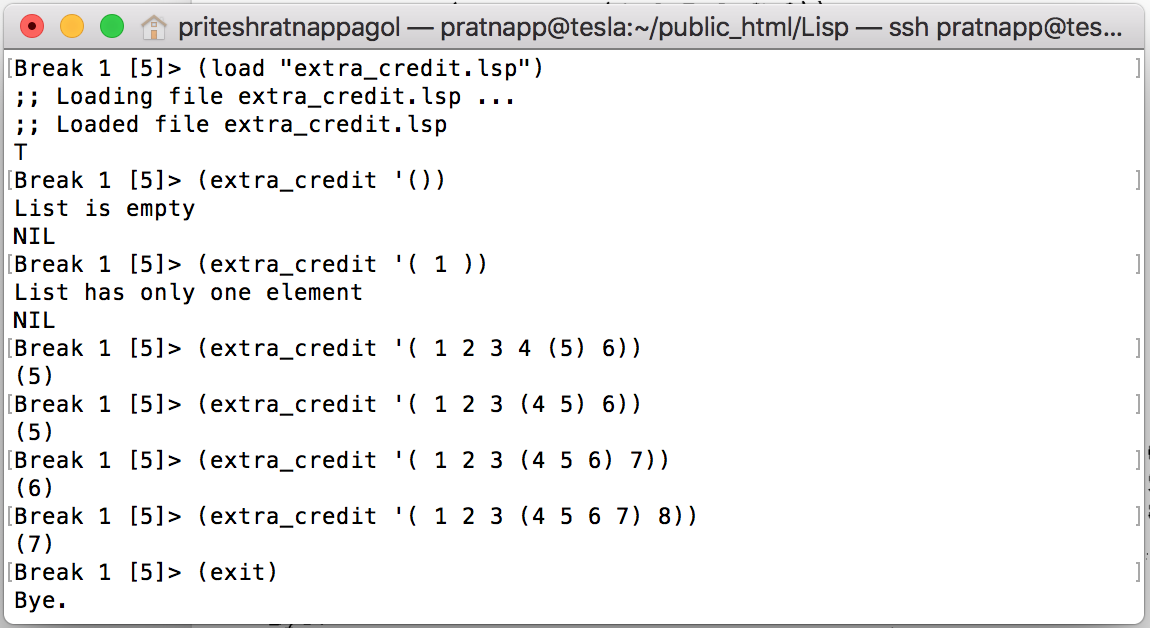


Screenshot 6. trace of shuffle operation on two lists

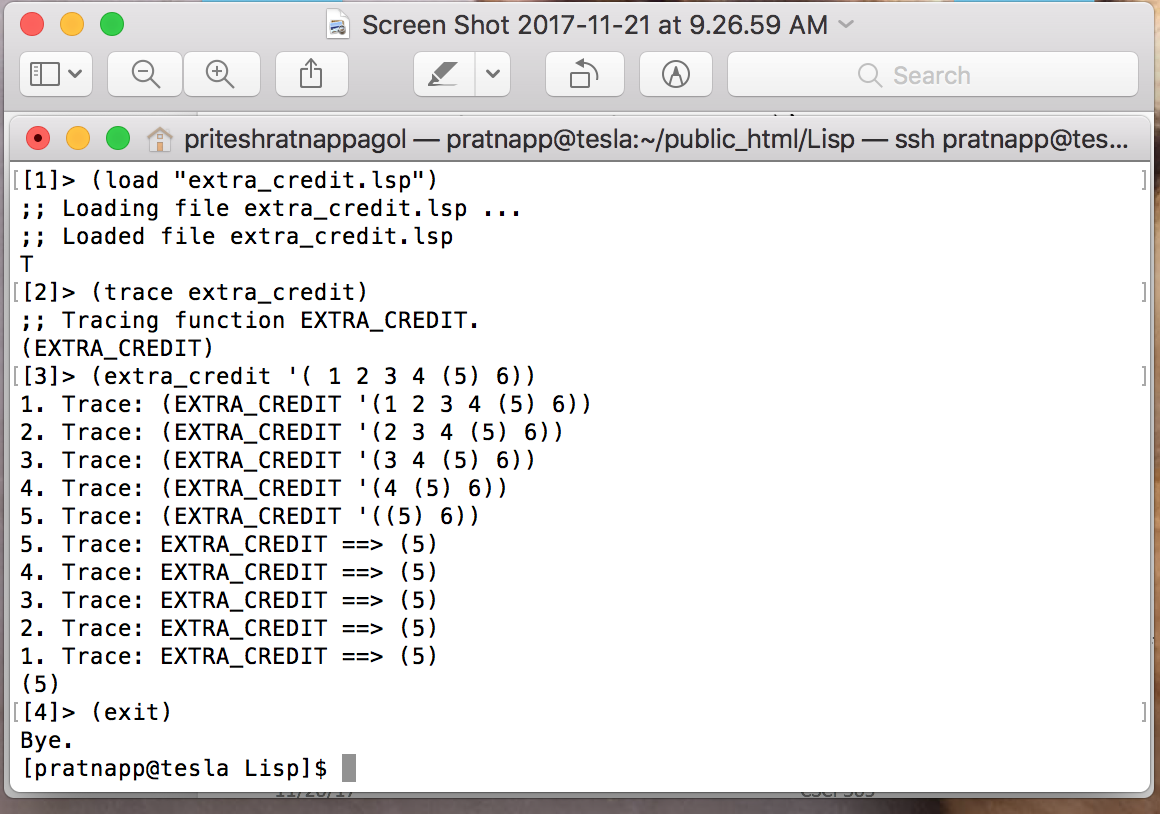
4)Extra Credit Example.

I have implemented it using these conditions:

1. Program first checks if list is empty and if so it returns the statement “List is empty” and exits the program
2. If list is not empty and contains only one atom in it, it returns “List has only one item” and exits the program
3. If list is not empty and contains two element, it returns the second last element by simply using the car function for the given list.
4. If list is not empty and contains more than two elements it recursively performs the above steps on tail of the list using cdr function till it is reached the step 3 and returns the second last element in the list.
5. If the second last element returned has the sub-list in it then it recursively performs the operation of finding the last element of that sub-list and returns that value.



Screenshot 7. Instance of finding last second element for extra credit



Screenshot 8. trace of finding last second element according to extra credit