

## CSCI330

### Ganzorig Arvinbileg

#### Homework 3

##### Question2.

1. Partition function:  

```
(defun partition (lst)
  (let ((mid (floor (/ (length lst) 2))))
    (list (subseq lst 0 mid) (subseq lst mid))))
```
2. Sort each half:  
Recursively apply Mergesort to each half
3. Merge Function:  

```
(defun merge (left right)
  (cond
    ((null left) right)
    ((null right) left)
    ((<= (car left) (car right))
     (cons (car left) (merge (cdr left) right)))
    (t (cons (car right) (merge left (cdr right))))))
```
4. Mergesort Function:  

```
(defun mergesort (lst)
  (if (<= (length lst) 1)
      lst
      (let ((halves (partition lst)))
        (merge (mergesort (first halves)) (mergesort (second halves))))))
```

##### Question4.

- a. When the insertion sort process starts:  
**Sorted items:** An empty list () because no elements have been sorted yet.  
**Unsorted items:** The original list to be sorted.  
When the insertion sort process ends:  
**Sorted items:** The fully sorted list in ascending order.  
**Unsorted items:** An empty list () because all elements have been moved to the sorted list.
- b. The process is trivially accomplished when the unsorted list is empty. At this point, the sorted list contains all the elements in the correct order, and no further work is needed.
  1. Take the first element from the unsorted list.
  2. Insert it into the correct position in the sorted list.
  3. Repeat the process with the remaining unsorted list.
- c. To insert an element into the sorted list, we need to:
  1. Track the items that have been examined (the part of the sorted list already processed).
  2. Track the item to be inserted.
  3. Track the items yet to be examined (the remaining part of the sorted list).

The insertion process can terminate in two ways:

1. The item to be inserted is smaller than the first element of the remaining sorted list. In this case, the item is inserted at the beginning.
2. The remaining sorted list is empty. In this case, the item is appended to the end of the sorted list.