```
# -*- coding: utf-8 -*-
"""
Created on Wed Oct 19 09:21:39 2016

@author:
ericgrimson
"""

def bubble_sort(L):
    swap = False
    while not swap:

 print('bubble sort: ' + str(L))
        swap = True
        for j in range(1, len(L)):

    if L[j-1] > L[j]:
                swap = False
                temp = L[j]

  L[j] = L[j-1]
                L[j-1] = temp

testList =
[1,3,5,7,2,6,25,18,13]

print('')
print(bubble_sort(testList))
print(testList)


def
selection_sort(L):
    suffixSt = 0
    while suffixSt != len(L):
        print('selection
sort: ' + str(L))
        for i in range(suffixSt, len(L)):
            if L[i] <
L[suffixSt]:
                L[suffixSt], L[i] = L[i], L[suffixSt]
        suffixSt += 1


testList = [1,3,5,7,2,6,25,18,13]


print('')
print(selection_sort(testList))
print(testList)


def merge(left, right):
    result
= []
    i,j = 0,0
    while i < len(left) and j < len(right):
        if left[i] <
right[j]:
            result.append(left[i])
            i += 1
        else:

result.append(right[j])
            j += 1
    while (i < len(left)):

result.append(left[i])
        i += 1
    while (j < len(right)):

result.append(right[j])
        j += 1
    print('merge: ' + str(left) + '&' + str(right) +
```

```
' to ' +str(result))
    return result

def merge_sort(L):
    print('merge sort: ' + str(L))

  if len(L) < 2:
        return L[:]
    else:
        middle = len(L)//2
        left =
merge_sort(L[:middle])
        right = merge_sort(L[middle:])
        return merge(left,
right)

testList = [1,3,5,7,2,6,25,18,13]

#print('')
#print(merge_sort(testList))
```