In [12]:

```
def bubblesort(list):
    for iter_num in range(len(list)-1,0,-1):
        for idx in range(iter_num):
            if list[idx]>list[idx+1]:
                temp = list[idx]
            list[idx] = list[idx+1]
            list[idx+1] = temp
list = [19,2,31,45,6,11,121,27]
bubblesort(list)
print(list)
```

```
[27, 121, 45, 45, 45, 19, 19, 19]
```

In [14]:

```
def merge_sort(unsorted_list):
     if len(unsorted_list) <= 1:
        return unsorted_list
# Find the middle point and devide it
   middle = len(unsorted_list) // 2
    left_list = unsorted_list[:middle]
    right_list = unsorted_list[middle:]

    left_list = merge_sort(left_list)
    right_list = merge_sort(right_list)
    return list(merge(left_list, right_list))

# Merge the sorted halves
def merge(left_half,right_half):
    res = []
    while len(left_half) != 0 and len(right_half) != 0:
        if left_half[0] < right_half[0]:
            res.append(left_half[0])
            left_half.remove(left_half[0])
        else:
            res.append(right_half[0])
            right_half.remove(right_half[0])
    if len(left_half) == 0:
        res = res + right_half
    else:
        res = res + left_half
        return res
unsorted_list = [64, 34, 25, 12, 22, 11, 90]
print(merge_sort(unsorted_list))
```

```
  File "<tokenize>", line 5
    middle = len(unsorted_list) // 2
    ^
IndentationError: unindent does not match any outer indentation level
```

In [20]:

```
def insertion_sort(InputList):
   for i in range(1, len(InputList)):
      j = i-1
      nxt_element = InputList[i]
# Compare the current element with next one
   while (InputList[j] > nxt_element) and (j >= 0):
      InputList[j+1] = InputList[j]
      j=j-1
   InputList[j+1] = nxt_element
list = [19,2,31,45,30,11,121,27]
insertion_sort(list)
print(list)
```

```
[19, 2, 31, 45, 30, 11, 27, 121]
```

In [5]:

```
def linear_search(values, search_for):
   search_at = 0
   search_res = False
# Match the value with each data element
   while search_at < len(values) and search_res is False:
      if values[search_at] == search_for:
         search_res = True
      else:
         search_at = search_at + 1
   return search_res
l = [64, 34, 25, 12, 22, 11, 90]
print(linear_search(l, 12))
print(linear_search(l, 91))
```

```
True
False
```

In [10]:

```
for i in range(10):
    print(i,end =" ")
```

```
0 1 2 3 4 5 6 7 8 9
```

In [7]:

```
for i in range(3, 10, 2):
    print(i, end =" ")
```

```
3 5 7 9
```

In [ ]:

```
```