Sorting Notes

1 Merge Sort

```
def merge_sorted(xs):
    '''
    Returns the input list xs in sorted order.
    '''
    if len(xs) <= 1:
        return xs
    else:
        mid = len(xs) // 2
        left = xs[:mid]
        right = xs[mid:]
        return _merged(merge_sorted(left), merge_sorted(right))


def _merged(xs, ys):
    '''
    Assuming xs and ys are sorted lists,
    returns a sorted list containing the elements of both xs and ys.
    Runs in linear time.
    '''
```

2 Quick Sort

```
def quick_sorted(xs):
    '''
    Returns the input list xs in sorted order.
    '''
    if len(xs) <= 1:
        return xs
    mid = len(xs) // 2
    pivot = xs[mid]
    xs_smaller = [x for x in xs if x < pivot]
    xs_bigger  = [x for x in xs if x > pivot]
    xs_equal   = [x for x in xs if x == pivot]
    return quick_sorted(xs_smaller) + xs_equal + quick_sorted(xs_greater)
```