Quiz

1) Below is some code to perform quicksort. Create the method partition. (5 points)

```java
void quicksort(int[] array){
    quicksort(array, 0, array.length - 1);
}

void swap(int[] array, i, k){
    int t = array[i];
    array[i] = array[k];
    array[k] = t;
}

void quicksort(int[] array, int left, int right){
    if (left < right){
        int pivot = partition(array, left, right);
        quicksort(array, left, pivot-1);
        quicksort(array, pivot+1, right);
    }
}

int partition(int[] array, int leftmost, int rightmost){
    int pivot = array[leftmost];
    int pIndex = leftmost;
    for (int k = leftmost + 1; k <= rightmost; k++){
        if (array[k] < pivot){
            pIndex++;
            swap(array, k, pIndex);
        }
    }
    swap(array, leftmost, pIndex);
    return pIndex;
}
```
2) Which one of quicksort, heapsort, and mergesort is stable? Also, what does stable mean? (2 points)

Mergesort is stable.
A sorting algorithm is stable if two equal elements x and y are in the original array such that x has a lower index than y, then x will still precede y in the final, sorted array.

3) By hand, perform mergesort on the following list of numbers. Explain your steps, so that I can understand what you're doing. 5, 2, 8, 3, 9, 4, 1, 7 (3 points)