Brandon Kinard
Problem 2 PDF

```
void mergesort(int[] a, int arraySize) {
        int listsize, xsize;

        for (listsize = 1; listsize <= arraySize; listsize*=2) {
         for (int i = 0, j = listsize; (j+listsize) <= arraySize; i += (listsize*2), j += (listsize*2)) {
             int b[] = new int[arraySize];
           for (int x=0; x<arraySize; x++)
             b[x] = a[x];
          merge(b, listsize, listsize);
         }
        }
```

In the best case the algorithm will run extremely close to normal mergesort and the time complexity will be O(nlogn);
In the worst case the time complexity will be O(n^2logn). This will happen if the array is originally in reverse order. This is because it will take linear time to rearrange the sub arrays and linear time to merge them back together.