2b.

Merge sorting algorithms are, in best and worst case O(NlogN) algorithms, but the in place merge sort is stable because the elements with the same key appear in the same order in the sorted array as they did in the original array. The operation that causes the bottleneck is definitely the recursions that exist in the array splitting algorithm because it uses up a lot of memory and time in order to keep breaking down the array:

```
void mergeSort2 (int array2[ ], int low, int high){
        if (low < high) {
                int mid;
                mid = (high + low) / 2;
                mergeSort2(array2, low, mid);
                mergeSort2(array2, mid +1, high);

                mergeB(array2, low, high, mid);
}
```