```c
#include <stdio.h>
/* ================================================================================
 *                      File: msort.c
 *                      Purpose: Demonstrate Merge Sort
 *                      Author: Tamal Chakraborty
 * ================================================================================ */
void copy(int b[], int a[], int l, int r) {
        int i;
        int j = l;
        int n = r - l + 1;
        for (i = 0; i < n; i++) a[j++] = b[i];
}
/* ================================================================================ */
void merge(int a[], int l, int r, int m) {
        int b[r - l + 1];
        int i = l, j = m + 1, k = 0;
        while ((i <= m) && (j <= r)) {
                if (a[i] <= a[j]) b[k++] = a[i++];
                else b[k++] = a[j++];
        }
        int p;
        if (i > m) {
                for (p = j; p <= r; p++) b[k++] = a[p];
        }
        else {
                for (p = i; p <= m; p++) b[k++] = a[p];
        }
        copy (b, a, l, r);
}
/* ================================================================================ */
void mergesort(int a[], int l, int r) {
        if (l < r) {
                int m = (l + r) / 2;
                mergesort(a, l, m);
                mergesort(a, m + 1, r);
                merge(a, l, r, m);
        }
}
/* ================================================================================ */
int main(int argc, char** argv) {
        int a[argc - 2];
        int i = 0;
        for (i; i < argc - 1; i++) {
                a[i] = atoi(argv[i + 1]);
        }
        mergesort(a, 0, argc - 2);
        for (i = 0; i < argc - 1; i++) printf("%d ", a[i]);
        return 0;
}
```