**アルゴリズムとデータ構造　問題16-1のプログラム実装の考え方**

**(1) 入力例**

本プログラムの入力は最初の行に要素数 n を与え，続く n 個の整数値 a0～a(n−1) を空白または改行で区切って標準入力から与える形式であるため，例えば「5」「3 1 2 5 4」という与え方のほか「5」「3」「1」「2」「5」「4」のように行を分けてもよく，各値は 32bit 符号付き int の範囲で負数も許容され，n は 0 以上とする仕様であり，プログラムは scanf で順に読み取り，長さ n の動的配列に格納した後にマージソートを実行するための作業配列 buff を確保し，異常時（確保失敗や不正入力）には処理を中止する設計である．

**(2) 出力例**

出力は昇順に整列された配列要素を 1 行に空白区切りで表示した後に改行し，次の行に反転数（交換回数） mc を 10 進整数で表示する仕様であり，例えば入力が n=5，配列が「3 1 2 5 4」であれば出力は「1 2 3 4 5」に続けて改行，次行に「3」を出す（反転対は (3,1)，(3,2)，(5,4) の 3 組）という形になり，数値の末尾に余計な空白は付かず，各行末には改行が 1 つだけ入り，反転数は long long で計数されるため大きな入力にも対応できる実装である．

**(3) 入力に対する出力結果の妥当性の説明**

妥当性として，本実装は分割統治に基づく安定なマージソートを用い，左右に分割した配列をマージする際，右側の要素 a[i] が左側退避配列 buff の先頭要素より小さい場合に，左側に残存する要素数（p−j）だけ mc に加算することで「i<j かつ a[i]>a[j]」となる反転対の個数を漏れなく数え上げるため，理論的に出力 mc は配列中の全反転数と一致し，整列結果は自然昇順で正確かつ一意に定まり，計算量は O(n log n)，追加メモリは O(n) として課題要件を満たしていると言える．