

## アルゴリズムとデータ構造 問題 13-1 のプログラム実装の考え方

(1) 入力の仕様 本プログラムは最初の行で要素数  $n$  を読み取り、 $n$  が 0 以上 10000 以下であることを前提として、続く入力から空白または改行で区切られた  $n$  個の整数を順に読み込む仕様である。読み取りに失敗した場合や  $n$  が範囲外の場合は直ちに処理を終了し、不完全な配列を扱わないため安全である。整数の値域は C 言語の `int` の範囲に従い、負数を含んでもよい。入力例として「5」に続けて「3 1 4 1 5」を与えると、長さ 5 の配列  $\{3, 1, 4, 1, 5\}$  を確保して以降のヒープ処理に渡す動作となる。

(2) 出力の仕様 整列後の配列を第 1 行に昇順で空白区切りで出力し改行する。第 2 行には `downheap` 呼び出し回数 (`dhc`)、第 3 行には交換回数 (`sc`) を 10 進整数で 1 行ずつ出力する。 $n=0$  のときは第 1 行が空行となり `dhc=0`, `sc=0` が続く。 $n=1$  のときは唯一の要素のみを出力し、計数は 0 である。例として入力「5/3 1 4 1 5」に対しては「1 1 3 4 5」に続けて `dhc` と `sc` が表示され、いずれも実行時の堆積化と下方調整の回数に一致する。

(3) 妥当性の説明 実装は最大ヒープを構築してから根と末尾の要素を交換し、縮小した部分配列に対して `downheap` によりヒープ性を回復する手続きを末尾まで反復するヒープソートである。`build_max_heap` は下位の内部節点から順に `downheap` を呼び出して全体を最大ヒープにし、その後の各反復で最大値が配列末尾に確定するため、最終的に昇順列が得られることが保証される。計算量は構築  $O(n)$ 、抽出反復  $O(n \log n)$  で合計  $O(n \log n)$  となり、出力された `dhc` と `sc` はそれぞれ下方調整と交換の回数を正しく計測しており、仕様要件を満たしている。