

問題 4-3 のプログラム実装の考え方

学籍番号 22059 氏名 来間 空 提出日 2025 年 5 月 30 日

(1) 入力例

本プログラムへの入力は、操作終了コード 6 が与えられるまで標準入力へ逐次供給される整数列 $\langle op, v \rangle$ であり、 $op=1$ (探索), $op=2$ (挿入), $op=3$ (削除) では直後に探索・挿入・削除対象キー v (32 ビット符号付き整数範囲) を伴い、 $op=4$ (前順出力)・5 (全解放)・6 (終了) では追加引数を取らない。各整数は ASCII 半角空白または改行で区切り、余分な空行や文字を含めない。例として「2 30 2 10 2 20 1 10 4 3 10 4 6」を入力すると 30,10,20 の順に挿入後 10 を探索・出力し、次に 10 を削除して再出力し、最後に終了する。

(2) 出力例

出力は操作に応じて順次生成され、 $op=1$ では探索成功時“YES”，失敗時“NO”， $op=4$ では空木なら“NONE”，非空なら根→左→右の前順で列挙したキー列を半角空白区切りで一行表示し行末に LF のみを付加する。他の操作は正常終了時に何も出力しない。前掲例では探索結果として“YES”が一行目出力され、続く $op=4$ で“30 10 20 ”，削除後の $op=4$ で“30 20 ”がそれぞれ改行付きで表示され、出力中に説明文や余分な空白が追加されることはないため自動評価に適合する。

(3) 入力に対する出力結果の妥当性の説明

挿入・探索・削除はいずれも二分探索木の順序条件 $left < root < right$ を保った再帰実装であり、削除時に左部分木最大値ノードを繰り上げることで構造を維持するため、各操作後に `printNode` で得られる前順列が理論的な木状態と常に一致し、探索結果の YES/NO もキー存在性と整合する。例で削除前に表示される“30 10 20 ”と削除後の“30 20 ”は期待通りであり、重複挿入が無視される仕様や `freeTree` によるメモリ解放も確認できることから、本プログラムは問題要件（正確性、 $O(h)$ 時間性能、メモリリーク無）を満たしていると判断できる。