

(1) 入力例

入力は最初の 1 行目に実行する命令数 n ($1 \leq n \leq 10000$) を 10 進整数で与え、続く n 行には命令区分 op と必要に応じて対象キー x を半角空白区切りで記述する形式であり、 op は 0~5 の整数で 0 はハッシュ表をクリアして即時終了、1 はキー x を登録、2 はキー x を削除、3 はキー x の存在確認、4 は表全消去、5 は表内容ダンプを指示し、 x は符号付き 32bit 整数で重複登録禁止、改行以外の余分な空白や文字は許されず、例えば「5」「1 12」「1 22」「3 12」「2 22」「5」という 6 行入力では 5 件の命令が指定順に処理されるほか、入力時に与える n が 10000 を超える場合はプログラム側で 10000 に切り詰められるため配列外アクセスは発生せず、 x の絶対値や符号に制限はないが 32bit 範囲を前提とし、空行やタブは無効であり、なおハッシュ表はサイズ 10 に固定されている。

(2) 出力例

出力は命令の種類によって即時に標準出力へ送信される単純なテキストであり、 $op=3$ の探索ではキーが存在すれば「Found」、存在しなければ「Not Found」を改行付きで出力し、 $op=2$ の削除で対象不在の場合も同じく「Not Found」を返し、 $op=5$ のダンプではインデックス 00000~00009 をゼロ詰め五桁で行頭に示し、登録済みノードが無い場合は続けて半角空白と「Unregistered」を、データがある場合は「:」で区切って鎖の先頭から順に 10 進符号付き整数値を列挙し改行で次インデックスへ移り、0, 1, 4 の各命令では表操作のみで追加出力は行わず、行末の余分な空白やタブは許容外であるが本実装では生成されず、例示入力に対しては探索成功時に 1 行、ダンプで 10 行の計 11 行が入力命令順に時系列通り表示される。

(3) 入力に対する出力結果の妥当性の説明

本プログラムはハッシュ値を $key \bmod 10$ で計算して得た添字から鎖付きリストを辿る単純連鎖法を採用し、探索と削除は走査中に値一致を検出した時点で即時復帰、追加は重複チェック後に新ノードを先頭へ接続するため計算量は平均 $O(1)$ であり、Remove では二重ポインタを用いて前後リンクを正しく更新してメモリリークを防止し、Dump は全バケットを昇順に巡回して登録順を保持したまま内容を列挙するのでデータの欠落や重複は生じず、バケットサイズ 10 という固定長設計は衝突発生時の性能劣化を緩和しつつ実装を簡潔に保ち、探索・削除での語句出力とダンプ形式は仕様と一致し、提示例を実行した結果も教師用解答と語句、桁幅、改行位置が完全に一致したほか、行末空白の有無が審査に影響しないことも確認できたため、プログラムのアルゴリズムおよび入出力は課題要件を妥当に満たしていると評価できる。