

ハッシュの基本

1 導入

この講義で最重要なのは、ハッシュを「全体を順番に探す」のではなく、「値から置き場所を素早く決める」仕組みとして見ることです。探索で遅くなりやすいのは、候補を1つずつ見ることです。ハッシュでは、値から添字のような場所を計算して、見る候補を最初から絞ります。

2 用語と定義

ハッシュとは、値を一定範囲の整数へ写す関数です。衝突とは、異なる2つの値が同じハッシュ値になることです。

3 方針

まず「どこに置くか」をハッシュ値で決めます。そのあと、衝突したときにどう処理するかを考えます。

4 直感的な説明

図書館で本を名前順に1冊ずつ探すより、分類番号で棚を先に絞れたほうが速いです。ハッシュもそれと似ていて、値から「まず見るべき場所」を決めます。

5 厳密な説明

5.1 1. ハッシュ表

配列の添字を使って値を管理するのがハッシュ表です。値 x に対してハッシュ関数 $h(x)$ を計算し、その場所を調べます。

5.2 2. 衝突

異なる値でも同じ場所へ送られることがあります。これが衝突です。したがって、ハッシュは「1回で必ず見つかる」のではなく、「平均的に見る候補を少なくできる」仕組みです。

5.3 3. 計算量

平均的には探索・追加・削除が $O(1)$ に近いですが、衝突が多いと遅くなります。だからハッシュ関数と表の大きさの設計が重要です。

6 見分け方

- 順序は不要で、「ある値があるか」を速く知りたいならハッシュを疑います。
- 大小関係が必要ななら、ハッシュより木やソートのほうが向くことがあります。

7 最終形

hash = 値から場所を決めて候補を絞る仕組み

8 一言でいうと

- ハッシュは、全探索を避けて「まずどこを見るか」を決めるためのデータ構造です。