線形探索

1 導入

線形探索で最重要なのは、**配列に特別な構造がないなら、候補を 1 つずつ消していくしかない**と見抜くことです。

整列も索引もない探索では、「どこかをまとめて捨てる」根拠がありません。だから左から順に見て、調べ終わった要素を候補から外していく、という方針になります。この講義では、その発想をはっきり言語化します。

2 用語と定義

まず主役になる用語をそろえます。

線形探索 (Linear search) とは、先頭から順番に要素を比較して目標値を探す方法です。

配列 (Array) を $A = (A_0, A_1, \ldots, A_{n-1})$ とします。ここで n は要素数です。

目標値 (Target) を x とします。やりたいことは、$A_i = x$ となる添字 i を見つけることです。

比較回数 (Number of comparisons) とは、$A_i = x$ のような判定を何回したかを表す量です。

3 方針

配列に順序の情報がないなら、1 回の比較で分かるのは「いま見ている 1 個の要素が x かどうか」だけです。ここが出発点です。

したがって、探索の核心は「左から順に調べ、一致しなかった要素を候補から外す」と考えることです。

この見方をはっきりさせるために、各段階で「まだ見ていない区間だけが候補である」という不変条件を追います。

4 導出

4.1 1. なぜ 1 個ずつ見るのか

最初は、配列のどの位置に x があるか分かりません。しかも整列していないので、たとえば $A_5 > x$ だったとしても、右側や左側をまとめて捨てることはできません。

つまり 1 回の比較で除外できるのは、今見ている 1 個だけです。ここから、$A_0, A_1, A_2, \ldots$ の順に確認する方針が自然に出ます。

4.2 2. 不変条件を立てる

i 個の要素を調べ終えた時点で、$A_0, A_1, ..., A_{i-1}$ には x がなく、もし x が存在するなら候補は $A_i, A_{i+1}, ..., A_{n-1}$ だけである、と考えます。

これが線形探索の不変条件です。

- 開始時、まだ 1 個も見ていないので、候補は配列全体です。
- つぎに A_i を見ます。
- もし $A_i = x$ なら、その場で探索成功です。
- もし $A_i \neq x$ なら、A_i は候補から外れます。したがって次の時点では、候補は $A_{i+1}, A_{i+2}, ..., A_{n-1}$ だけになります。

このようにして、1 回比較するたびに候補をちょうど 1 個ずつ減らせます。

4.3 3. 終了条件と計算量

どこかで $A_i = x$ が見つかれば、その添字 i を返して終了です。

最後まで一致がなければ、候補は空になります。つまり $i = n$ まで進んだ時点で、「この配列には x がない」と結論できます。

したがって、最悪では n 回の比較が必要です。x が先頭にあれば 1 回で終わりますが、末尾にあるか、存在しなければ n 回かかります。よって計算量は $O(n)$ です。

5 どこまで成り立つか

線形探索は、配列が整列していなくても使えます。これは大きな長所です。必要なのは、要素を順番に取り出して比較できることだけです。

その代わり、1 回の比較で捨てられる候補は 1 個しかありません。したがって、要素数が大きいと探索時間は長くなります。

つまり、前提条件がほとんどいらない代わりに、計算量は $O(n)$ から改善しません。

6 最終形

$i = 0, 1, 2, ..., n-1$ の順に $A_i = x$ を調べ、最初に一致した添字を返します。

最後まで一致しなければ、「存在しない」と結論します。

したがって線形探索は

未確認の要素だけを候補として、左から 1 つずつ消していく探索

であり、計算量は $\boxed{O(n)}$ です。

7 一言でいうと

- 線形探索では、1 回の比較で捨てられる候補は 1 個だけです。
- だから、未確認の部分を候補集合として持ち、左から 1 個ずつ消していくと考えます。
- この候補集合の減り方が、線形探索の本体です。

8 関連リンク

→ 講義 **二分探索** lecture information algorithm
https://study.bem130.com/lecture/information/algorithm/search/二分探索-講義/