

アルゴリズム論 (第10回)



岩手県立大学
Iwate Prefectural University

佐々木研(情報システム構築学講座)

講師 山田敬三

k-yamada@iwate-pu.ac.jp

探索



岩手県立大学
Iwate Prefectural University

線形探索

- 配列a
 - 要素 $a[i]$ ($i=0,1,\dots,n-1$)
- 配列aから値xを探索
 - $i=0\sim n-1$ まで, 以下を繰り返す.
 - $a[i]$ とxを比較
 - $a[i]==x$ なら終了

二分探索(binary search)

- 前提条件
 - 要素 $a[i]$ ($i=0,1,\dots,N-1$) について $a[0]<a[1]<\dots<a[N-1]$ が成立
- 配列 a から x の場所を求める方法
 - x が配列 a にあるか否か, 配列のどの要素よりも大きいか, 小さいかも返値とする
- 返値
- $$i = \begin{cases} 0 & x \leq a[0] \\ j & a[j-1] < x \leq a[j] \quad (0 < j \leq N-1) \\ N & a[N-1] < x \end{cases}$$

二分探索(binary search)

アルゴリズム (P.82~83)

1. $x \leq a[0]$ または $x > a[n-1]$ であるかどうかを調査する。
そうでなければ、 $a[0] < x \leq a[n-1]$ の場合のみを考える。
2. left と right について初期化する
 $left = 0, right = n - 1$
3. 探索する配列の中央を探す
 $middle = (left + right) / 2$
4. x と中央の値 $a[middle]$ を比較する
 1. x が $a[middle]$ 以下の場合は、新しい探索配列の右側を $middle$ にする
 2. x が $a[middle]$ よりも大きい場合は、新しい探索配列の左側を $middle$ にする
5. 3~4 を $right - left \leq 1$ になるまで繰り返す。

計算量

- 検索範囲は比較する段階で元の配列の1/2となる
- 比較する回数 k は、全体の要素数 N とすると
$$2^k = N$$
すなわち
$$k = \log_2 N$$
- よって、
$$O(\log N)$$

使うときの注意点

- 検索結果*i*を用いて必ず配列を確認する
 - アルゴリズムは*n*を返すことがあるが、配列は*a[n-1]*までしか値がない
 - *x*が配列*a*の中に見つかるとは限らない。すなわち、*x == a[i]*とは限らないので、検査する必要がある。