

二分探索

境界を高速に探索する

kya (@tsaayk)

TMU-CS B4

April 30, 2022

目次

- はじめに
- 二分探索
- 実装
- 例題

はじめに

- 以下の問題が出てきたらどのようにして解きますか？

問題文

長さ N の広義単調増加列 $A = A_0, \dots, A_{N-1}$ が与えられます. また整数 K が与えられます. $A_i \geq K$ であるようなインデックス i のうち最小のものを出力してください. そのような i が存在しない場合は -1 を出力してください.

制約

◆ $0 \leq N, K \leq 10^5$

◆ $0 \leq A_i \leq 10^9$

はじめに

- 愚直に解く場合
 - 数列を先頭から見て行って、 $A_i \geq K$ であるような i があればそれを出
力する
 - 時間計算量は $O(N)$
 - これでも十分間に合う
- じゃあ K が複数与えられてそれぞれについて i を求める場合
は？
 - K が Q 個与えられるとき、時間計算量は $O(NQ)$
 - Q のサイズ次第では間に合わない
 - 数列 A は単調増加列なので、二分探索が使える

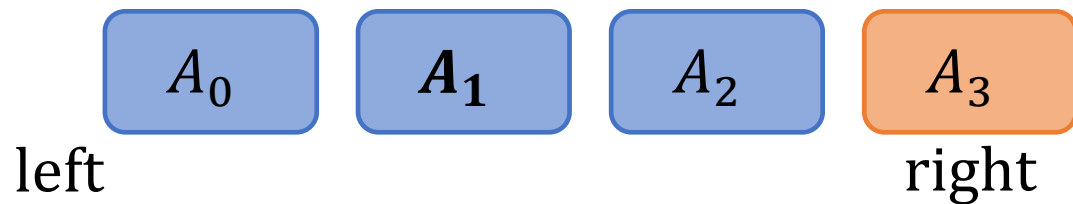
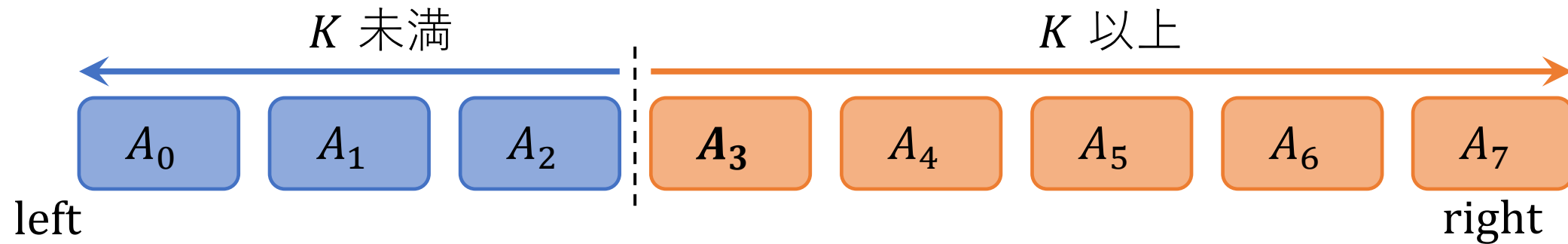
二分探索

- 二分探索とは
 - 単調性のある列に対して, 探索範囲を半分に絞り込む操作を繰り返すことで, 高速に目的の値を検索するアルゴリズム
- $O(\log(N))$ で答えを求められる
- 数列が単調性を持っていることが条件

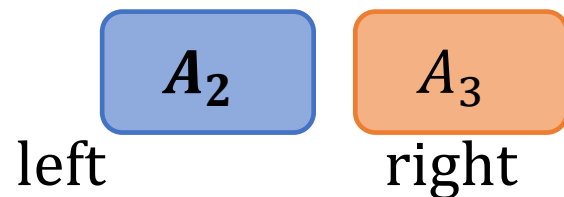
二分探索

- $\text{left}, \text{right}$ を探索する区間の左端と右端として初期化する
- $\text{mid} = (\text{left} + \text{right})/2$ として $A_{\text{mid}} < K$ の場合は $\text{left} = \text{mid}$, $A_{\text{mid}} \geq K$ の場合は $\text{right} = \text{mid}$ で更新することを繰り返す
 - $\text{right} - \text{left} > 1$ である間上記操作を繰り返す
- 上記の操作が終了したとき, left または right が $A_i \geq K$ であるような最小のインデックスとなる

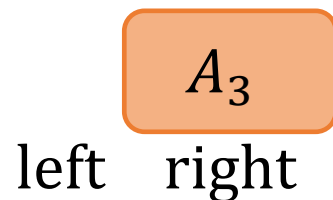
二分探索



→ A_4 が K 以上なので right を更新



→ A_1 が K 未満なので left を更新



→ A_2 が K 未満なので left を更新

実装：C++

- 区間 $(-1, N - 1]$ を探索する
 - left と right の間が K になる
 - 今回は「 K 以上」なので right が答えになるように左開右閉区間
 - 「 K 以下」の場合は左閉右开区間
 - K を見つけたい場合はどちらでも大丈夫

```
int left = -1, right = n - 1;
while (abs(right - left) > 1) {
    int mid = (left + right) / 2;
    if (a[mid] < k) left = mid;
    else right = mid;
}
```


実装：Python

- $A_i \geq K$ となる i が存在しない場合,
left = $n - 2$, right = $n - 1$ となって終了するので、最後にチェックするか、
あらかじめ $A_{N-1} \geq K$ であることを調べておく必要がある

```
left, right = -1, n - 1
while abs(right - left) > 1:
    mid = (right + left) // 2
    if a[mid] < k:
        left = mid
    else:
        right = mid
```

実装：テクニック

- 探索範囲を半開区間で考える癖をつけると実装が楽になる
 - 閉区間で実装すると全て値を満たしているときに、バグる可能性がある
 - `left, right` を `ok, ng` という変数名で実装している人も多い
 - 条件を満たす, 満たさない値を示す
- **While** 文の終了条件を常に $|\text{left} - \text{right}| > 1$ にしておくと, どんな問題でも使えるので, バグらせにくくなる
 - 探索区間が実数領域の場合は使えないので注意
 - 実数値の二分探索は $|\text{left} - \text{right}| > \text{EPS}$ にする
 - EPS: 自分で定義した限りなく小さい値 (10^{-6} など)
 - `left == right` はバグる
 - 詳しくは浮動小数点数の誤差などで検索

例題

典型アルゴリズム問題集 A - 二分探索の練習問題

- スライドの初めに出てきた問題
- 二分探索で解いてみてください
- 回答例 C++ : <https://atcoder.jp/contests/typical-algorithm/submissions/31326450>
- 回答例 Python : <https://atcoder.jp/contests/typical-algorithm/submissions/31326512>

例題

ABC246 D - 2-variable Function

- 非負整数 a, b を用いて $a^3 + a^2b + ab^2 + b^3$ と表せる N 以上の数の最小値を求める問題
- a, b 両方を全探索すると間に合わないが, 片方を固定すると単調性が見えてくる
- 「パラメータが複数ある場合はどれか一つを固定する」も競プロ頻出のテクニック
- 回答例 C++ : <https://atcoder.jp/contests/abc246/submissions/31326951>
- 回答例 Python : <https://atcoder.jp/contests/abc246/submissions/31327086>

例題

競プロ典型 90 問 007 - CP Classes (★3)

- 配列 $A = \{A_0, \dots, A_{N-1}\}$ について $|A_i - b|$ の最小値を求める問題
- b に最も近い値が求められればよい
- A をソートすると, b に近い値は二分探索で求められる
- 回答例 C++ : <https://atcoder.jp/contests/typical90/submissions/31327387>
- 回答例 Python : <https://atcoder.jp/contests/typical90/submissions/31327501>

例題

ARC050 B - 花束

- 手持ちの赤い花と青い花で最大何個の花束が作れるかを求める問題
- 花束を K 個作れるとき, 自明に花束を $K - 1$ 個作ることは可能
- 花束を作れる個数には単調性があるので, 「花束を K 個作れるか？」という判定問題を考える

例題

ABC023 D - 射撃王

- 二分探索の良問
- 風船を割る競技で得られる点数の最大値を求める問題
- 「最小値を最大化する問題」は多くの場合二分探索で解ける
- 「点数を K 点にできるか？」という判定問題を考える