

雇用計画 (Employment)

@joisino_

March 21, 2016

目次

問題概要

小課題 1 : 10 点 (累計 10 点)

小課題 2 : 30 点 (累計 40 点)

小課題 3 : 60 点 (累計 100 点)

得点分布

目次

問題概要

小課題 1 : 10 点 (累計 10 点)

小課題 2 : 30 点 (累計 40 点)

小課題 3 : 60 点 (累計 100 点)

得点分布

問題概要

- ▶ 数列が与えられる
- ▶ 以下のクエリを処理せよ
 - ▶ B_i 以上の数のみを考えたとき連続する部分の数を求めよ
 - ▶ 数列の値を変更せよ

サンプル入出力

8

6

3

5

4

サンプル入出力

8

採用

6

採用

3

不採用

5

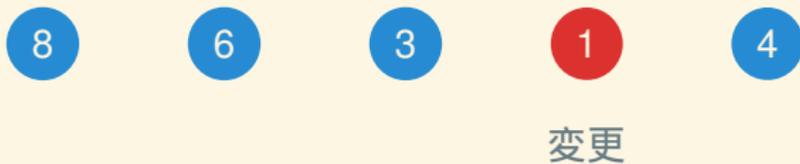
採用

4

不採用

- ▶ 評価値が 5 以上の候補者を採用する
- ▶ 連続部分は 2 個

サンプル入出力



- ▶ 4 番目の候補者の評価値を 1 に変更する

サンプル入出力

8

採用

6

採用

3

不採用

1

不採用

4

不採用

- ▶ 評価値が 5 以上の候補者を採用する
- ▶ 連続部分は 1 個

サンプル入出力

8

採用

6

採用

3

採用

1

不採用

4

採用

- ▶ 評価値が 3 以上の候補者を採用する
- ▶ 連続部分は 2 個

目次

問題概要

小課題 1 : 10 点 (累計 10 点)

小課題 2 : 30 点 (累計 40 点)

小課題 3 : 60 点 (累計 100 点)

得点分布

小課題 1 : 10 点 (累計 10 点)

▶ $N \leq 2000$

▶ $M \leq 2000$

小課題 1 : 10 点 (累計 10 点)

- ▶ シミュレーションをやるだけ
- ▶ クエリごとに誰が採用されるかを求めて、連続部分の個数を求める
- ▶ $O(NM)$ でできる

目次

問題概要

小課題 1 : 10 点 (累計 10 点)

小課題 2 : 30 点 (累計 40 点)

小課題 3 : 60 点 (累計 100 点)

得点分布

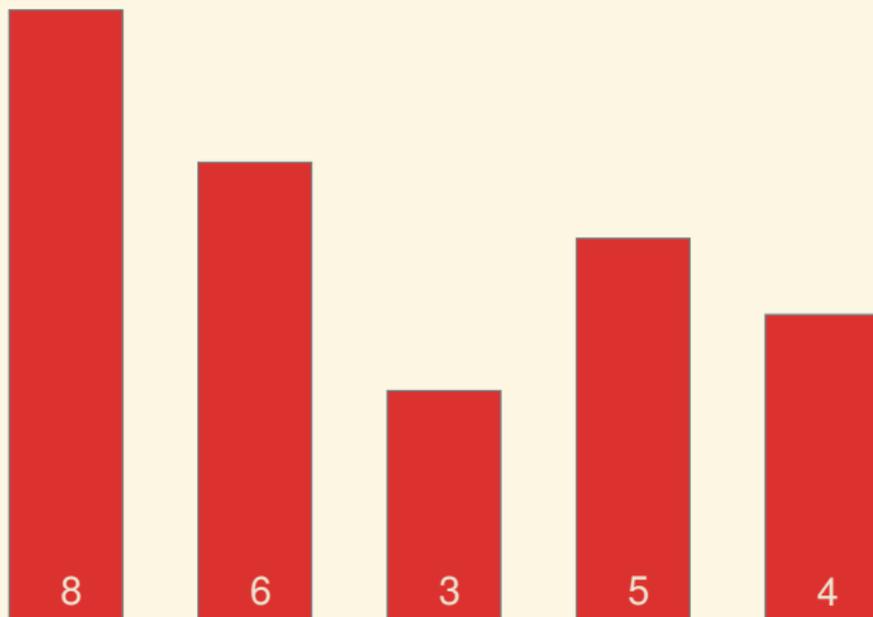
小課題 2 : 30 点 (累計 40 点)

- ▶ $T_i = 1$ (解答クエリのみ)
- ▶ $N \leq 200\,000$
- ▶ $M \leq 200\,000$

小課題2 : 30点 (累計40点)

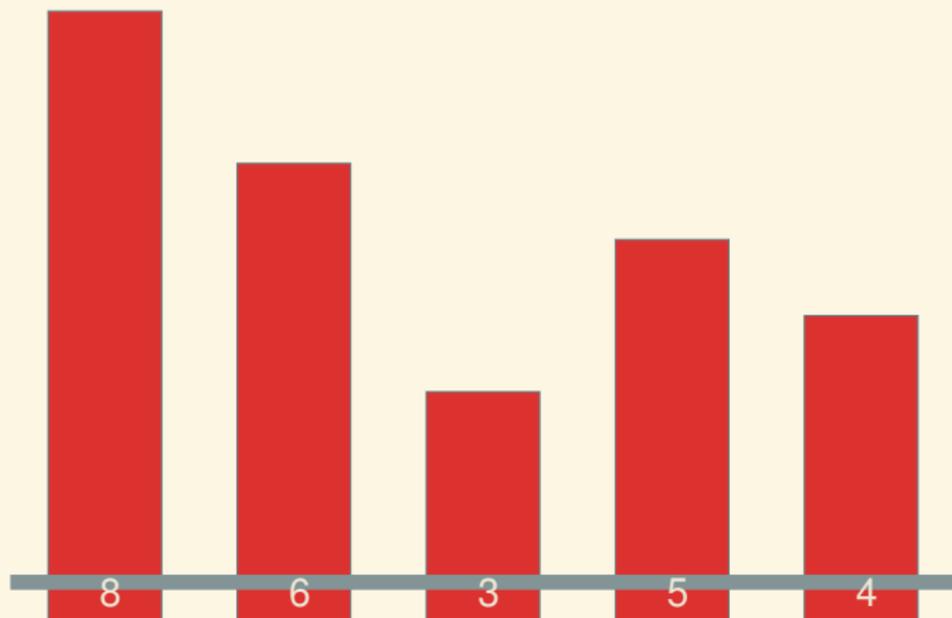
- ▶ 解答クエリを観察してみる

小課題2 : 30点 (累計40点)



- ▶ わかりやすい図
- ▶ 高さが評価の大きさを表す

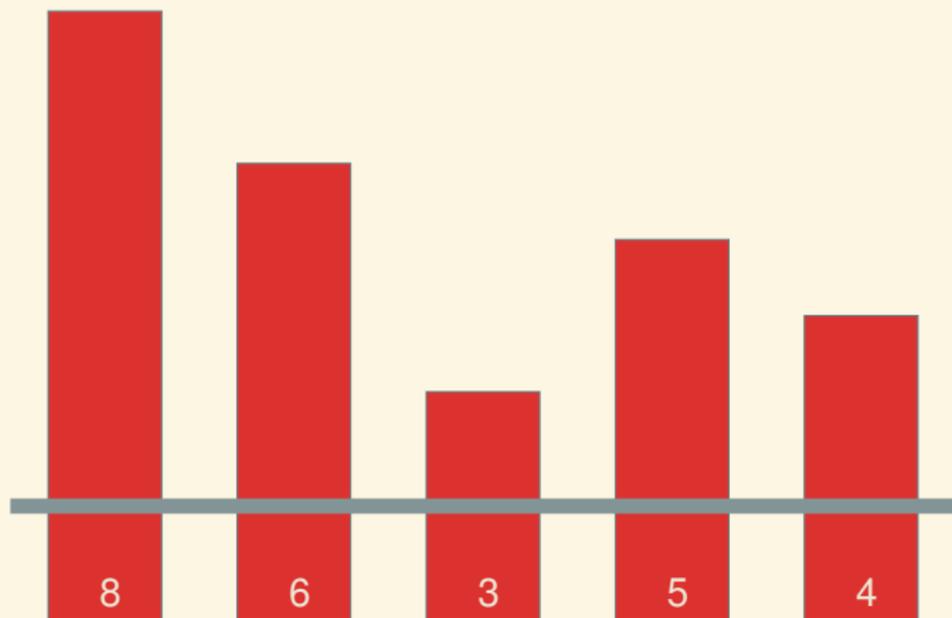
小課題 2 : 30 点 (累計 40 点)



▶ $B = 1$ のとき

▶ $ans = 1$

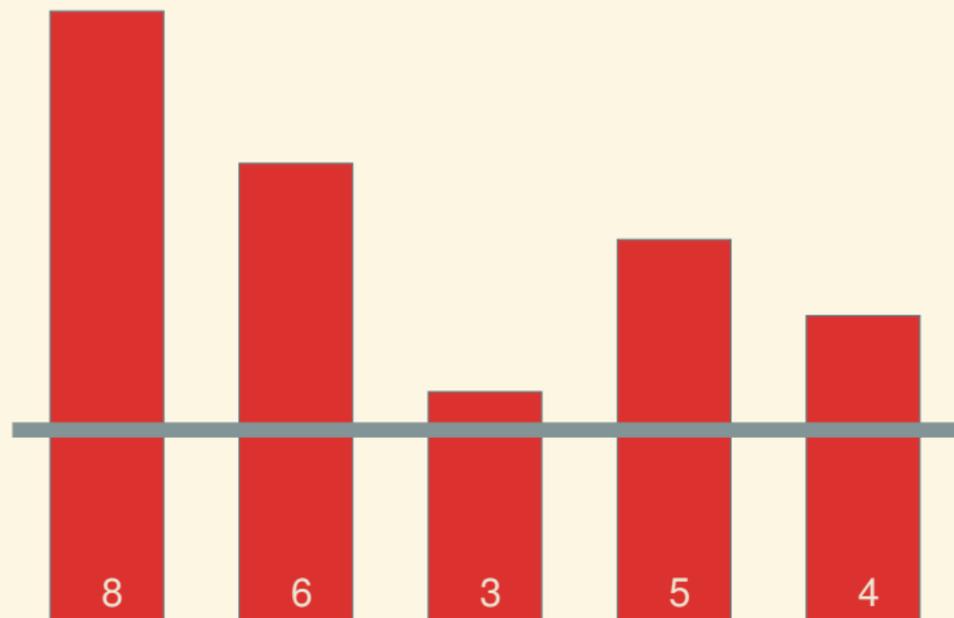
小課題 2 : 30 点 (累計 40 点)



▶ $B = 2$ のとき

▶ $ans = 1$

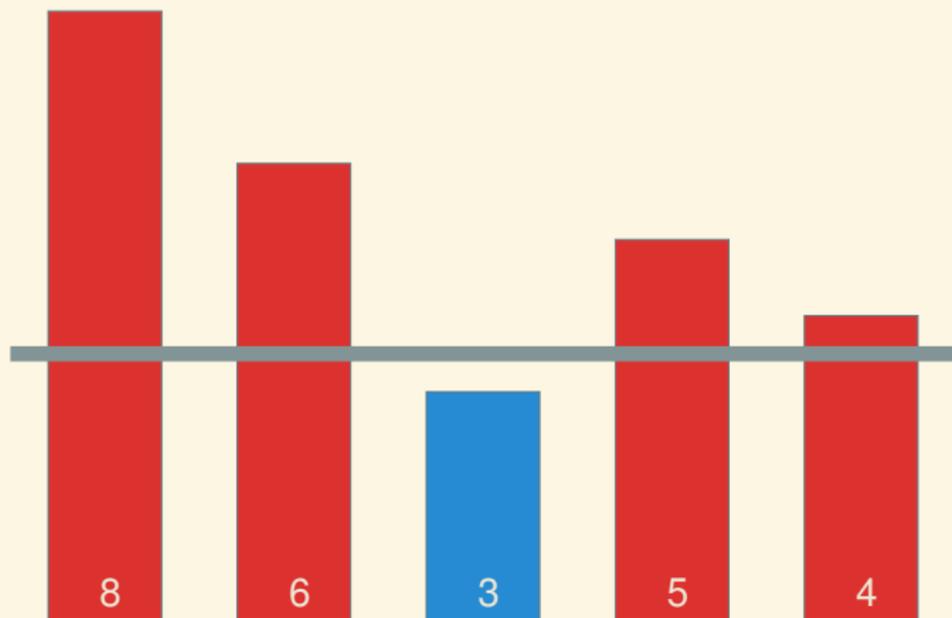
小課題 2 : 30 点 (累計 40 点)



▶ $B = 3$ のとき

▶ $ans = 1$

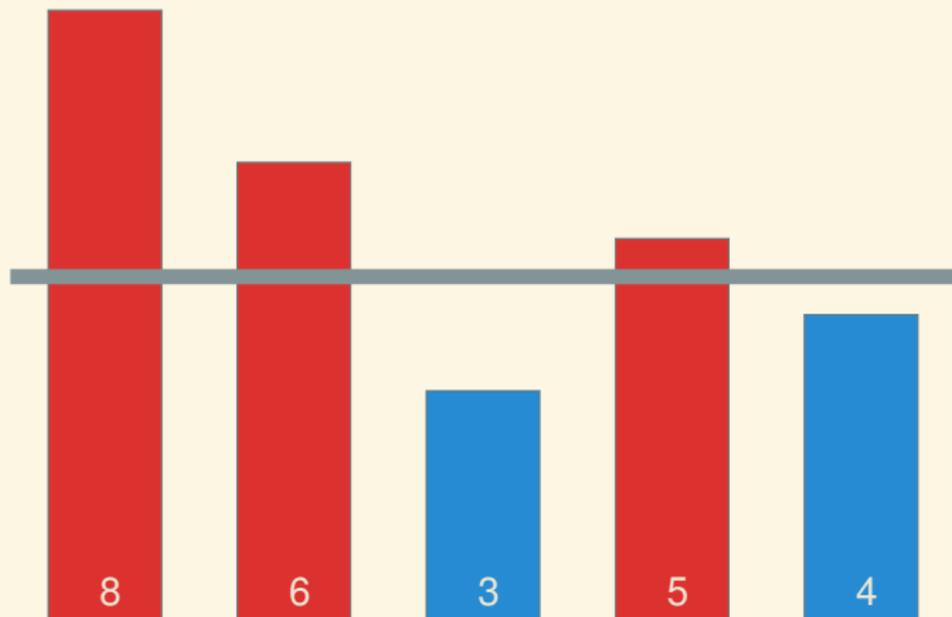
小課題 2 : 30 点 (累計 40 点)



▶ $B = 4$ のとき

▶ $ans = 2$

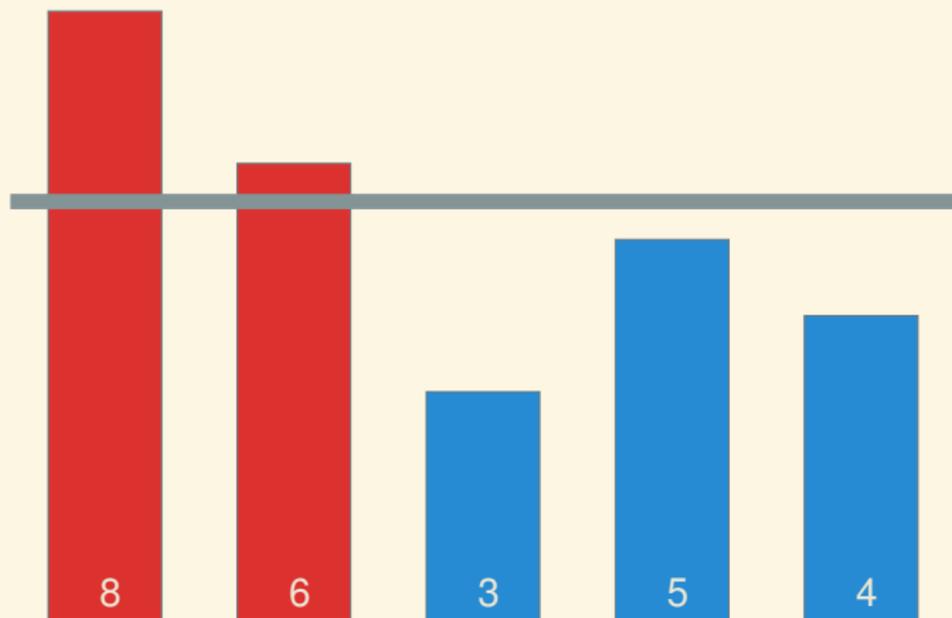
小課題 2 : 30 点 (累計 40 点)



▶ $B = 5$ のとき

▶ $ans = 2$

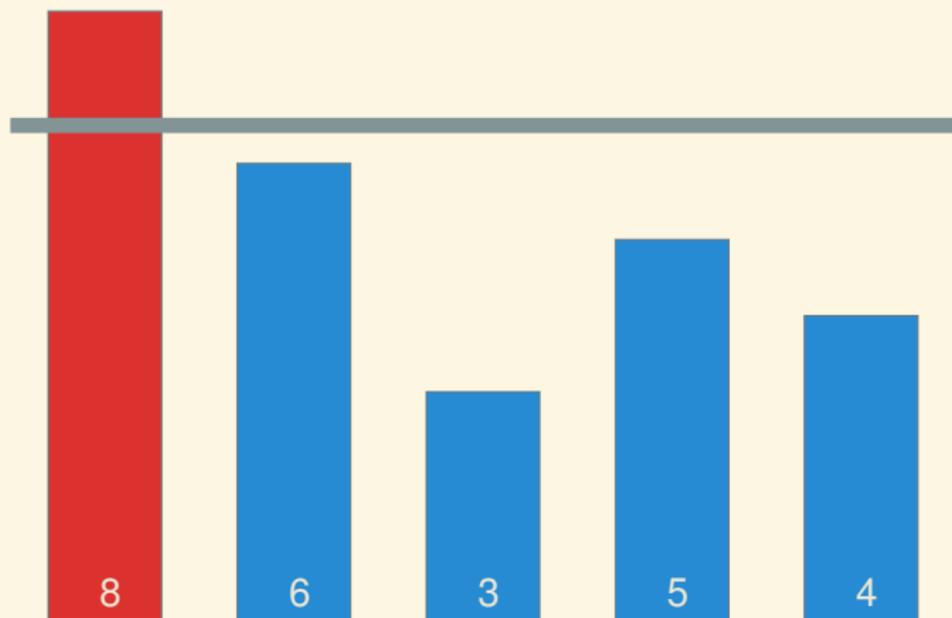
小課題 2 : 30 点 (累計 40 点)



▶ $B = 6$ のとき

▶ $ans = 1$

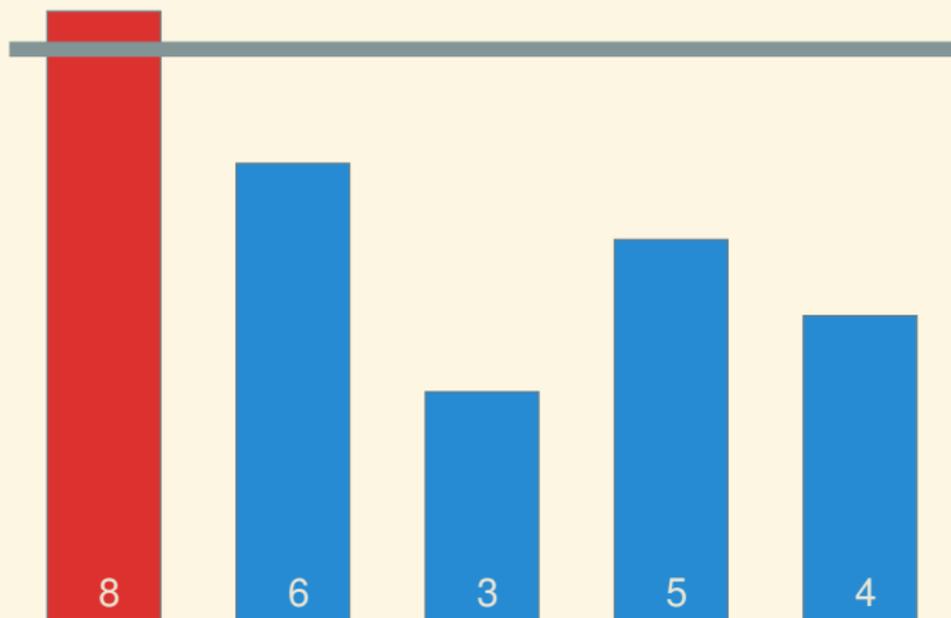
小課題 2 : 30 点 (累計 40 点)



▶ $B = 7$ のとき

▶ $ans = 1$

小課題 2 : 30 点 (累計 40 点)



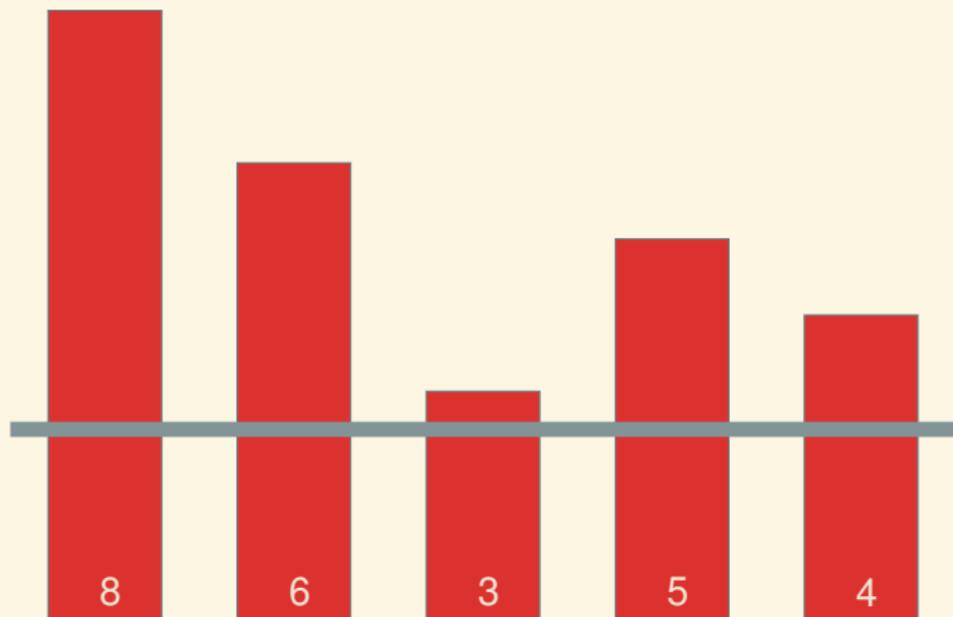
▶ $B = 8$ のとき

▶ $ans = 1$

小課題 2 : 30 点 (累計 40 点)

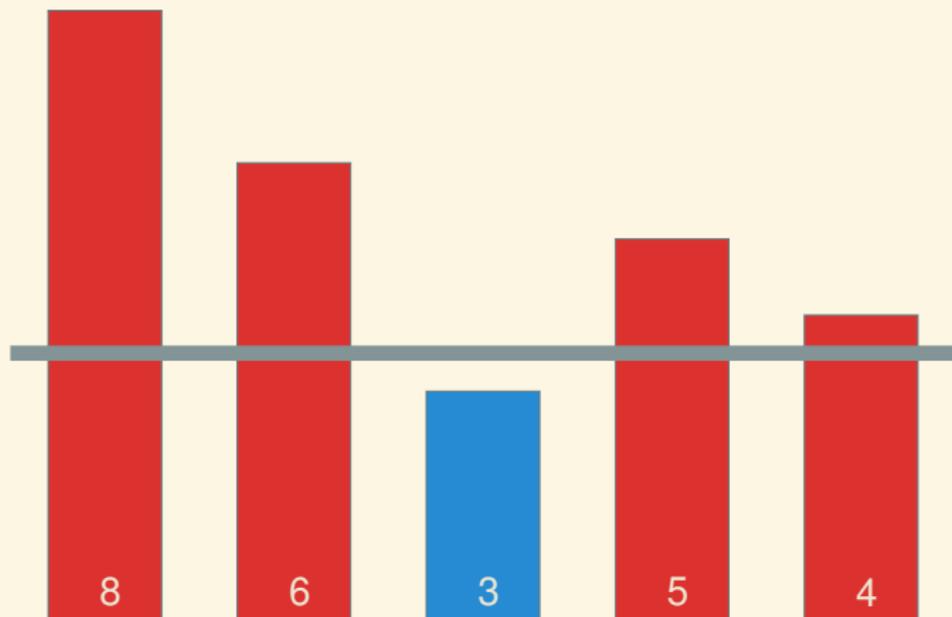
- ▶ 採用基準の小さい方からクエリをみていくと，採用される人は減っていく
- ▶ 採用される人が減ったときに答えがどう変わるかを計算すればよい

小課題2 : 30点 (累計40点)



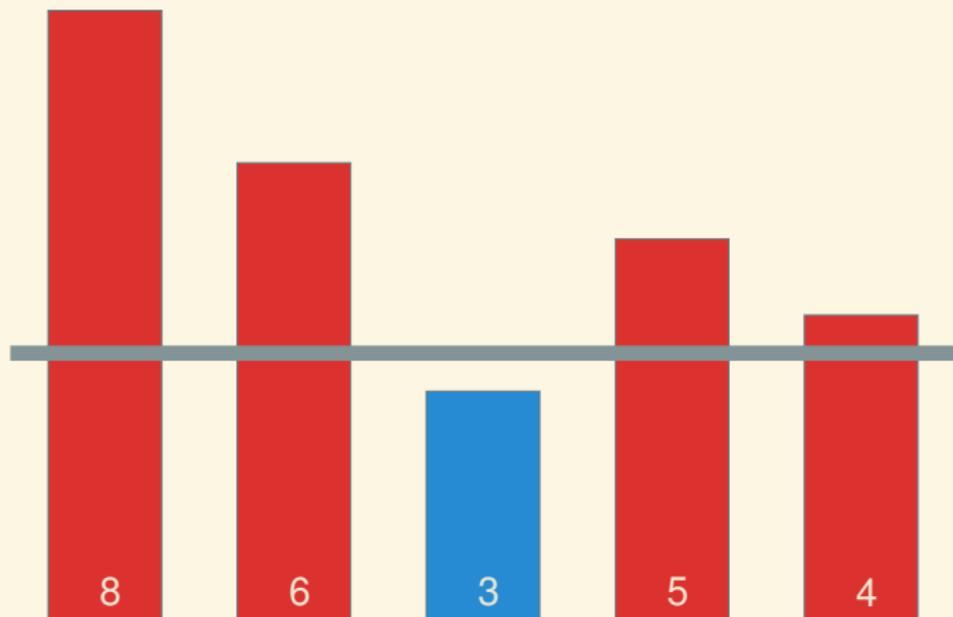
▶ 答えが増えるタイミング

小課題2 : 30点 (累計40点)



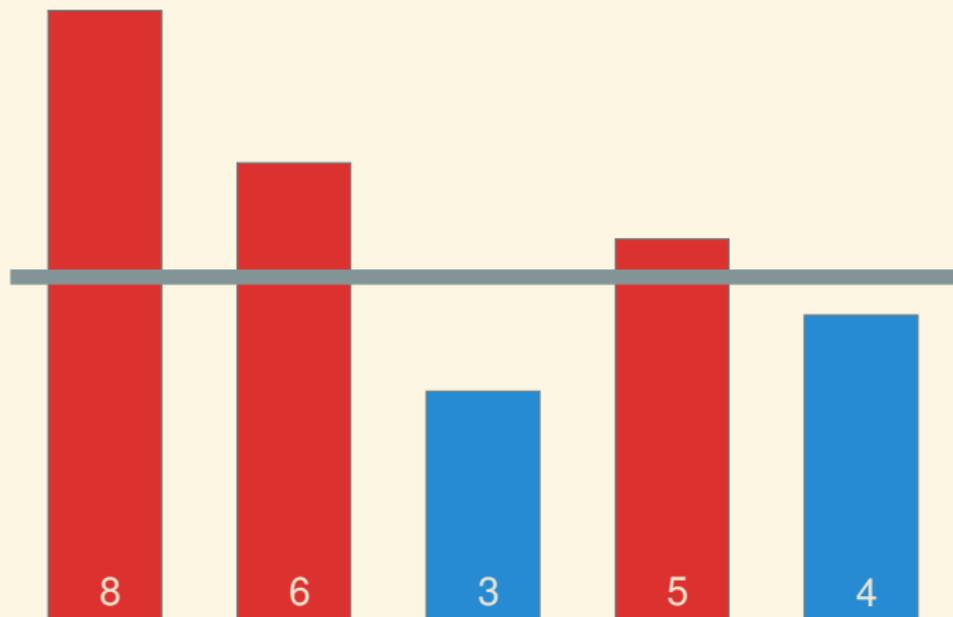
- ▶ 3人目が不採用になることで答えが1増えた

小課題2 : 30点 (累計40点)



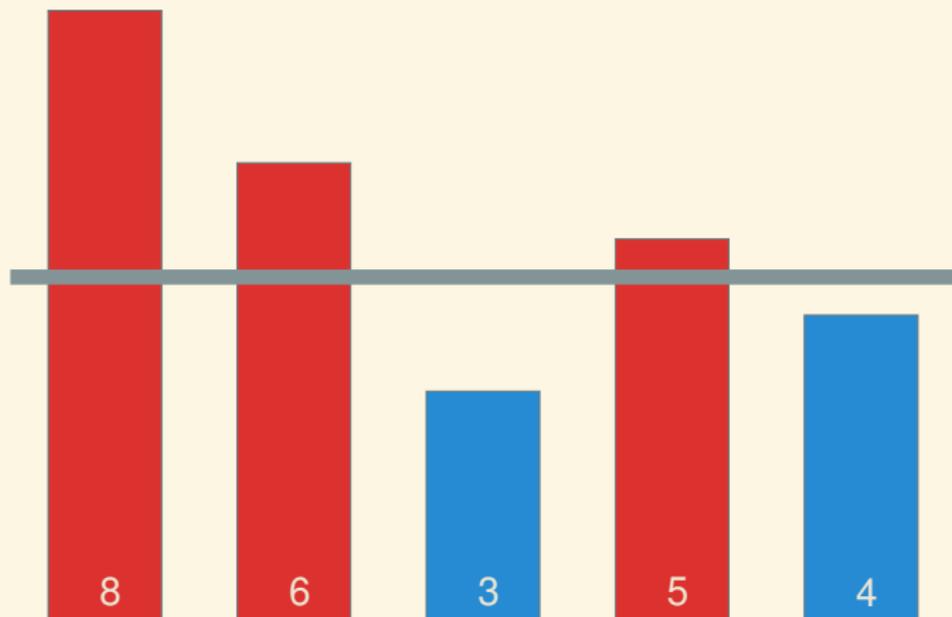
▶ 答えが変わらないタイミング

小課題2 : 30点 (累計40点)



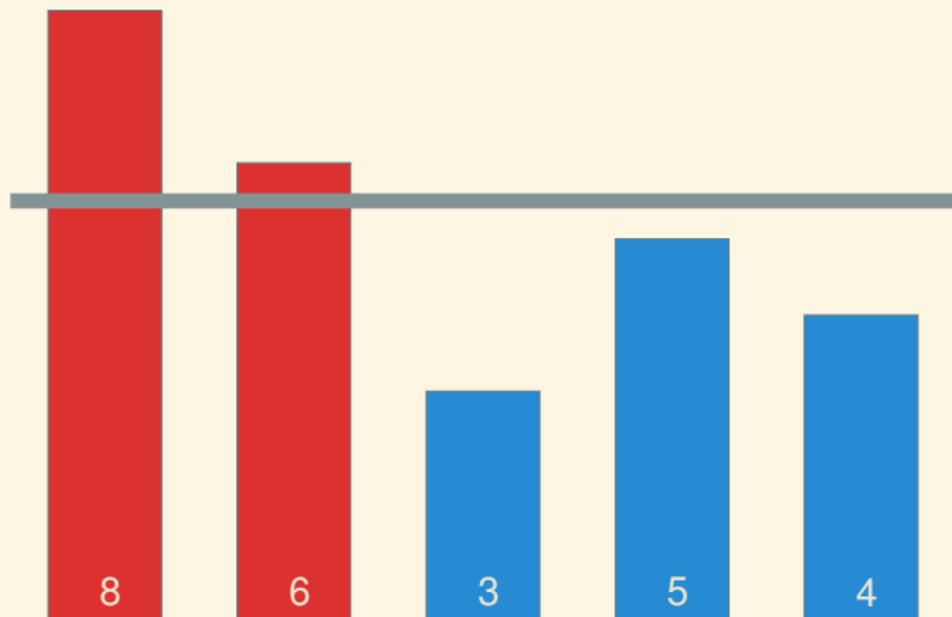
- ▶ 5人目が不採用になっても答えは変わらなかった

小課題2 : 30点 (累計40点)



▶ 答えが減るタイミング

小課題2 : 30点 (累計40点)



▶ 4人目が不採用になることで答えが1減った

小課題 2 : 30 点 (累計 40 点)

- ▶ 両隣が採用されているときに不採用になると、答えが 1 増える
- ▶ 両隣が不採用なときに不採用になると、答えが 1 減る
- ▶ 両隣のうち一方が採用，一方が不採用のときに不採用になると、答えは変わらない
- ▶ ただし 0 番目と $N + 1$ 番目には不採用の人がいるとする

小課題 2 : 30 点 (累計 40 点)

- ▶ クエリと候補者のソートに $O((N + M) \log(N + M))$
- ▶ 不採用になるのは合計 N 回 で毎回 $O(1)$ で処理できる .
- ▶ 答えが計算できているとクエリに答えるのは $O(1)$ ができる .
- ▶ よって , 全体の計算量は $O((N + M) \log(N + M))$
- ▶ 小課題 2 が解ける

目次

問題概要

小課題 1 : 10 点 (累計 10 点)

小課題 2 : 30 点 (累計 40 点)

小課題 3 : 60 点 (累計 100 点)

得点分布

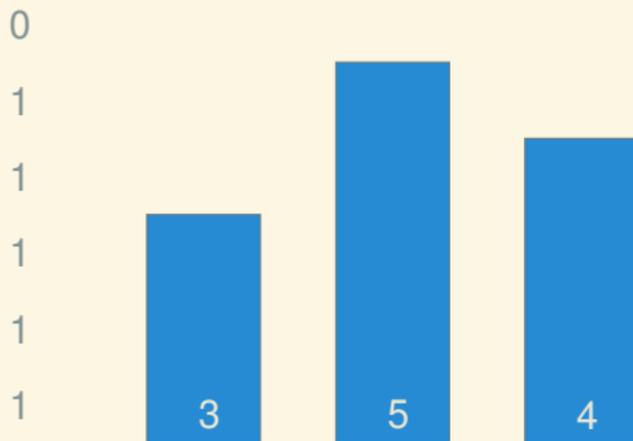
小課題3 : 60点 (累計 100点)

- ▶ 追加の制約はない
- ▶ $N \leq 200\,000$
- ▶ $M \leq 200\,000$

小課題3 : 60点 (累計 100点)

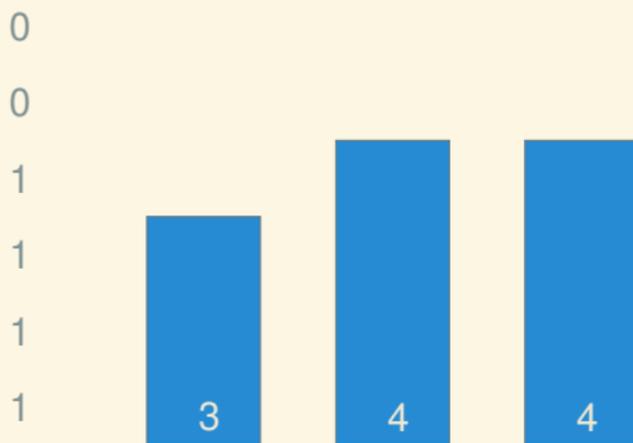
- ▶ 更新クエリを観察してみる

小課題3 : 60点 (累計100点)



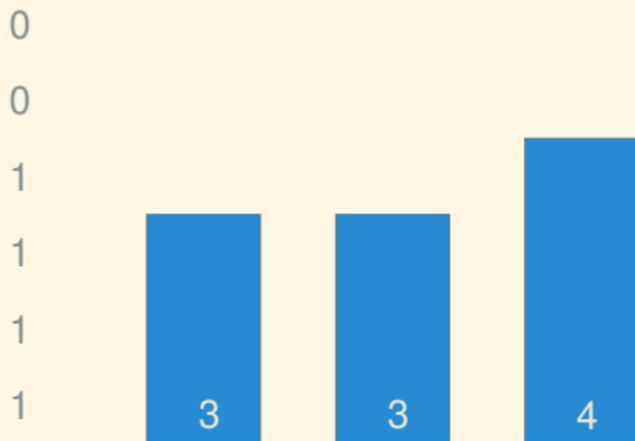
- ▶ 左側の数字はそこでの解答クエリの値
- ▶ 少しずつ真ん中の人の評価を減らしていきましょう

小課題3 : 60点 (累計 100点)



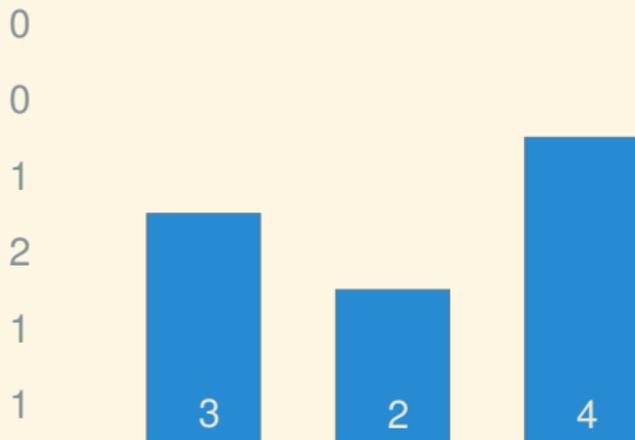
- ▶ 左側の数字はそこでの解答クエリの値
- ▶ 少しずつ真ん中の人の評価を減らしていきましょう

小課題3 : 60点 (累計 100点)



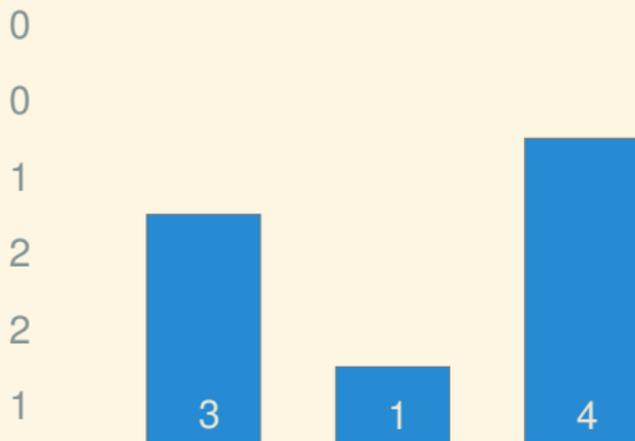
- ▶ 左側の数字はそこでの解答クエリの値
- ▶ 少しずつ真ん中の人の評価を減らしていきましょう

小課題3 : 60点 (累計100点)



- ▶ 左側の数字はそこでの解答クエリの値
- ▶ 少しずつ真ん中の人の評価を減らしていきましょう

小課題3 : 60点 (累計 100点)

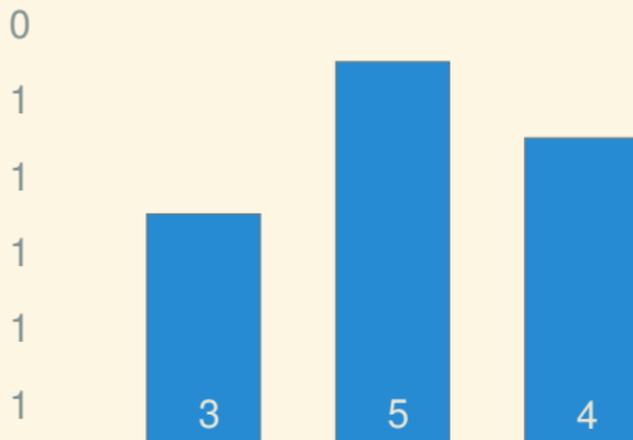


- ▶ 左側の数字はそこでの解答クエリの値
- ▶ 少しずつ真ん中の人の評価を減らしていきましょう

小課題 3 : 60 点 (累計 100 点)

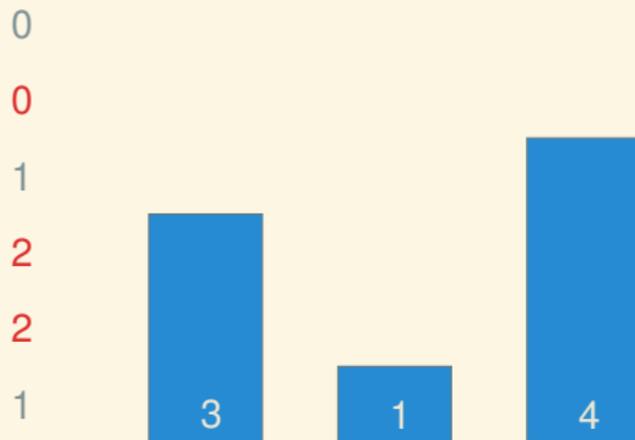
- ▶ 一人の評価を変更しても，解答クエリの値は高々 1 しか変わらない

小課題3 : 60点 (累計 100点)



▶ 一気に減らすとこんな感じ

小課題3 : 60点 (累計 100点)



▶ 一気に減らすとこんな感じ

小課題 3 : 60 点 (累計 100 点)

- ▶ どのように変化するのか
- ▶ 小課題 2 の考察と同様に考えると分かる
- ▶ 評価値が増加するときには，左右が共に存在するときに答えが 1 減少する
- ▶ 評価値が増加するときには，左右がどちらも存在しないときに答えが 1 増加する
- ▶ 評価値が減少するときはこの逆

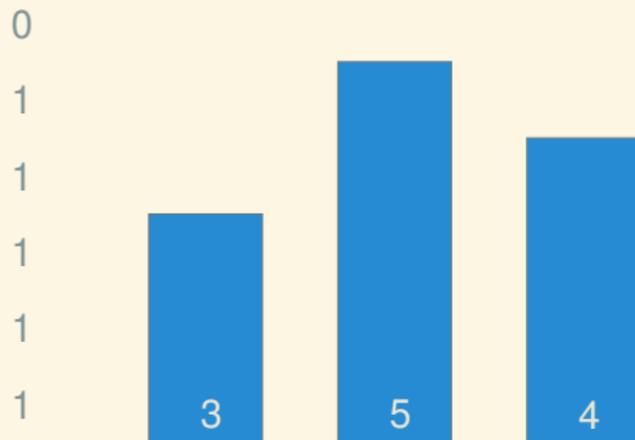
小課題3：60点(累計100点)

- ▶ 初期状態の答えを計算しておいて，更新クエリのたびにその答えをうまく変化させればよい
- ▶ 初期状態の答えを計算するのは，小課題2の方法でもできるし，全員の初期評価が0（このときの答えはどの位置でも0）で，それぞれの本当の初期評価になるように最初に更新クエリが飛んできたと考えると手間が省ける

小課題 3 : 60 点 (累計 100 点)

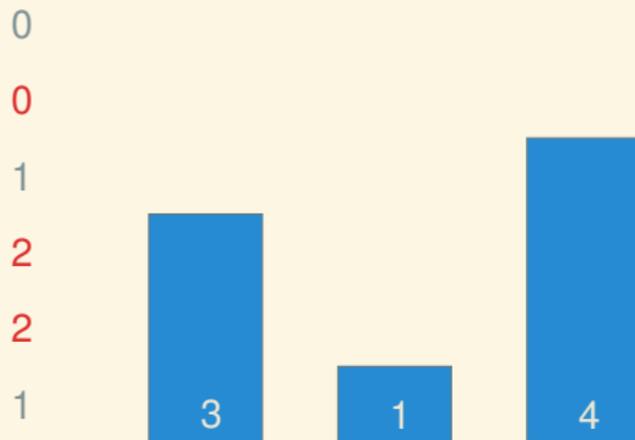
- ▶ 答えをうまく変更させるにはどうすればよいか
- ▶ 少し考えると, 更新クエリのたびに 1 増加する場所と 1 減少する場所はそれぞれ高々 1 つの区間であることが分かる
- ▶ 両隣の max より上と, min より下の部分

小課題3：60点(累計100点)



▶ 再掲

小課題3：60点(累計100点)



▶ 再掲

小課題3 : 60点 (累計 100点)

- ▶ 区間に1足したり1引いたりさえできれば...

小課題3 : 60点 (累計 100点)

Segment tree

小課題3：60点(累計100点)

- ▶ いろいろな方法で実装できる
- ▶ それぞれのノードに一様に加算された値を持ったり，BITを2つ使ったり（蟻本163ページ付近を参照）

小課題3 : 60点 (累計 100点)

- ▶ 今回は一点での値がわかれば良いので BIT 1 つでできる
- ▶ $[l, r)$ への x の加算は $\text{add}(l, x)$, $\text{add}(r, -x)$ を呼んで
- ▶ a での値を求めるには $\text{sum}(a)$ を呼べば良い
- ▶ 累積和を計算する感じ
- ▶ ただし, 評価値の最大は 10^9 なので, 最初に座標圧縮をするか, 動的に segtree を構築する必要がある

小課題 3 : 60 点 (累計 100 点)

- ▶ 座標圧縮に $O((N + M) \log(N + M))$
- ▶ 初期計算に $O(N \log(N + M))$
- ▶ クエリの処理に $O(M \log(N + M))$
- ▶ よって, 全体の計算量は $O((N + M) \log(N + M))$
- ▶ 小課題 3 が解ける
- ▶ 満点

小課題3 : 60点 (累計 100点)

- ▶ 別解
- ▶ クエリの平方分割
- ▶ 更新クエリが溜まったら答えを再計算し直す
- ▶ 解答クエリでは, 計算しておいた答えに溜まっている更新クエリを加味して答えを出す
- ▶ 全体の計算量は $O((N + M)(\log(N + M) + \sqrt{M}))$
- ▶ この解法だと少し実行時間制限が厳しいので, いろいろバケットサイズを試したり高速化する必要があるかもしれない

目次

問題概要

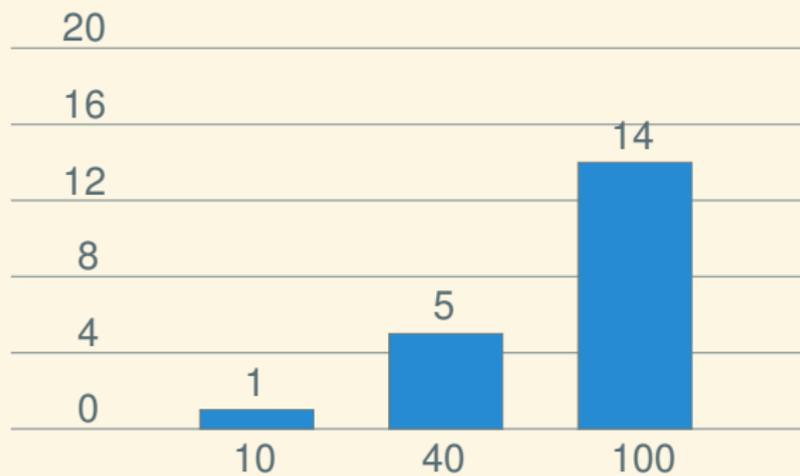
小課題 1 : 10 点 (累計 10 点)

小課題 2 : 30 点 (累計 40 点)

小課題 3 : 60 点 (累計 100 点)

得点分布

得点分布



▶ 優秀

小課題3 : 60点 (累計 100点)

- ▶ ご清聴ありがとうございました