Donuts プロコンチャレンジ2015

解説

株式会社Donuts　久野慎弥（@kuno4n）

おつかれさまでした！

問題A　ドーナツの体積

問題A

- ドーナツの体積を求めましょう

問題A　解答

- (R*R*π)*(D*2*π)　を計算すればOK
-

問題A　解答

- (R*R*π)*(D*2*π)　を計算すればOK
- πは、算術ライブラリの定数、acos(-1)、自分で書く、といった方法

問題B　Tokyo 7th シスターズ

問題B

- 9≦n≦16 人のアイドルから9人選んで、ユニットの能力値を最大化させよう

問題B　解答

- **n**が小さいので、各アイドルを使う・使わないを**全列挙**できます
-

問題B　解答

- **n**が小さいので、各アイドルを使う・使わないを**全列挙**できます
- （定石）**2**進数で管理

問題B　解答

```
for(i = 0; i < 2^n; i++){



}
```

問題B　解答

```
for(i = 0; i < 2^n; i++){
  if(iのビットが立っている数が9個){


  }
}
```

問題B　解答

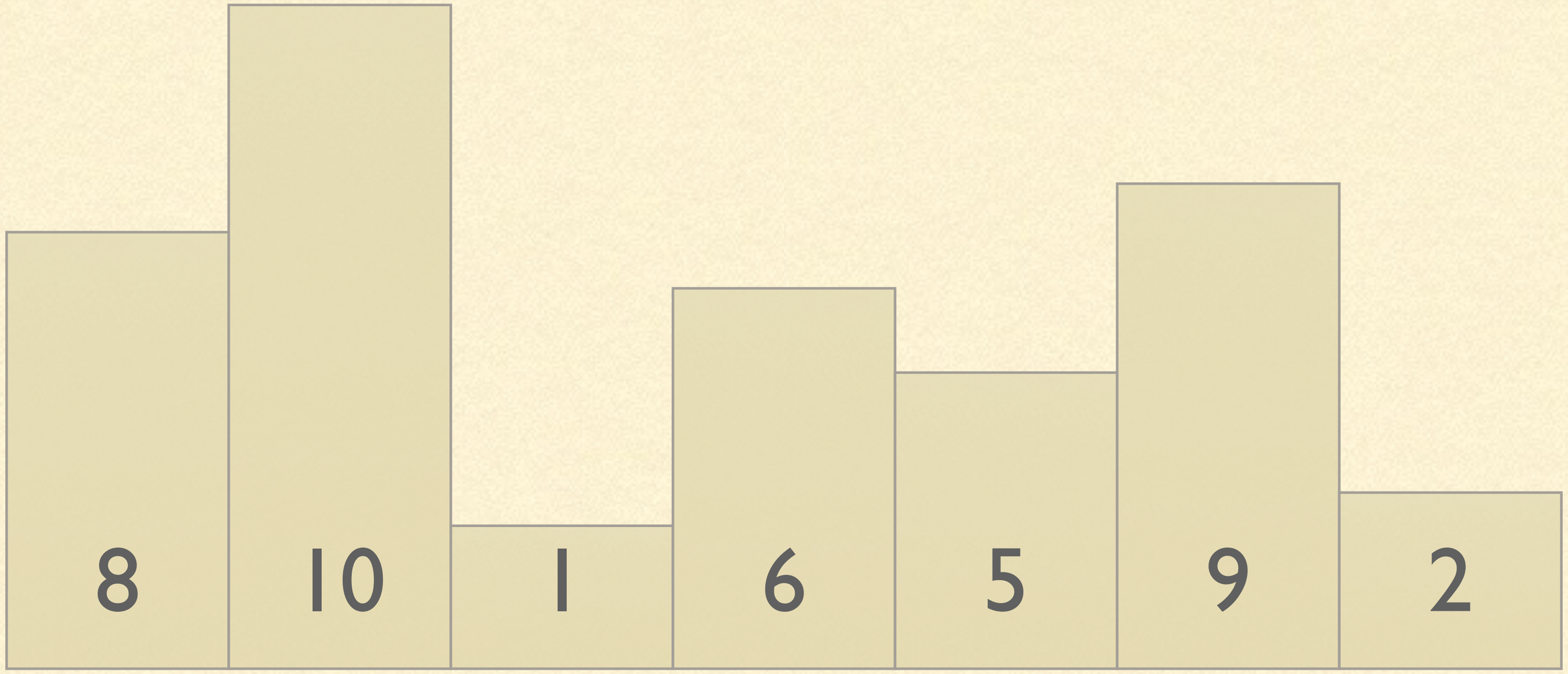
```
for(i = 0; i < 2^n; i++){
    if(iのビットが立っている数が9個){

        // 結果が大きければ更新

    }
}
```

問題C　行列のできるドーナツ屋

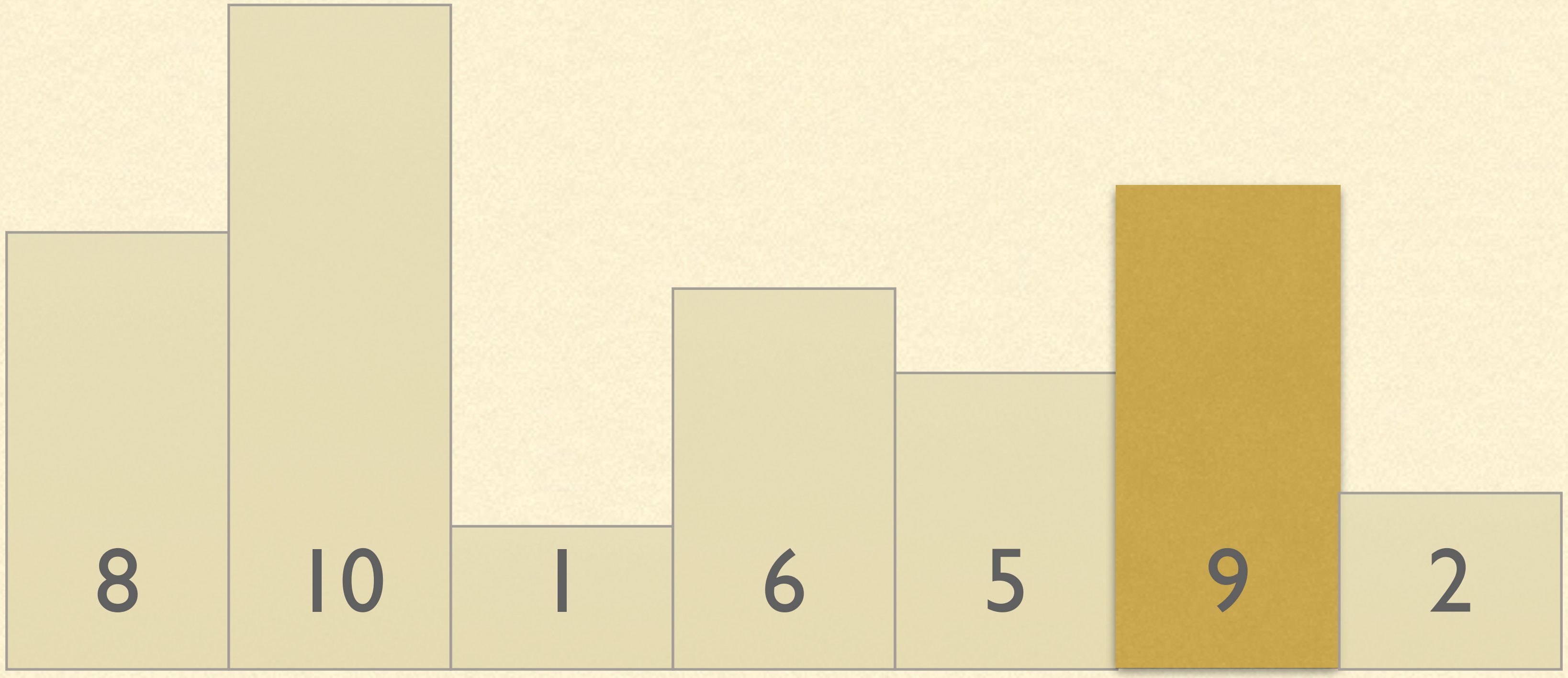
問題C

- 一列に並ぶ1≦n≦100000 人の身長が与えられます
- 各人の「前を見た時に見える人の数」を求めて下さい

問題C　例

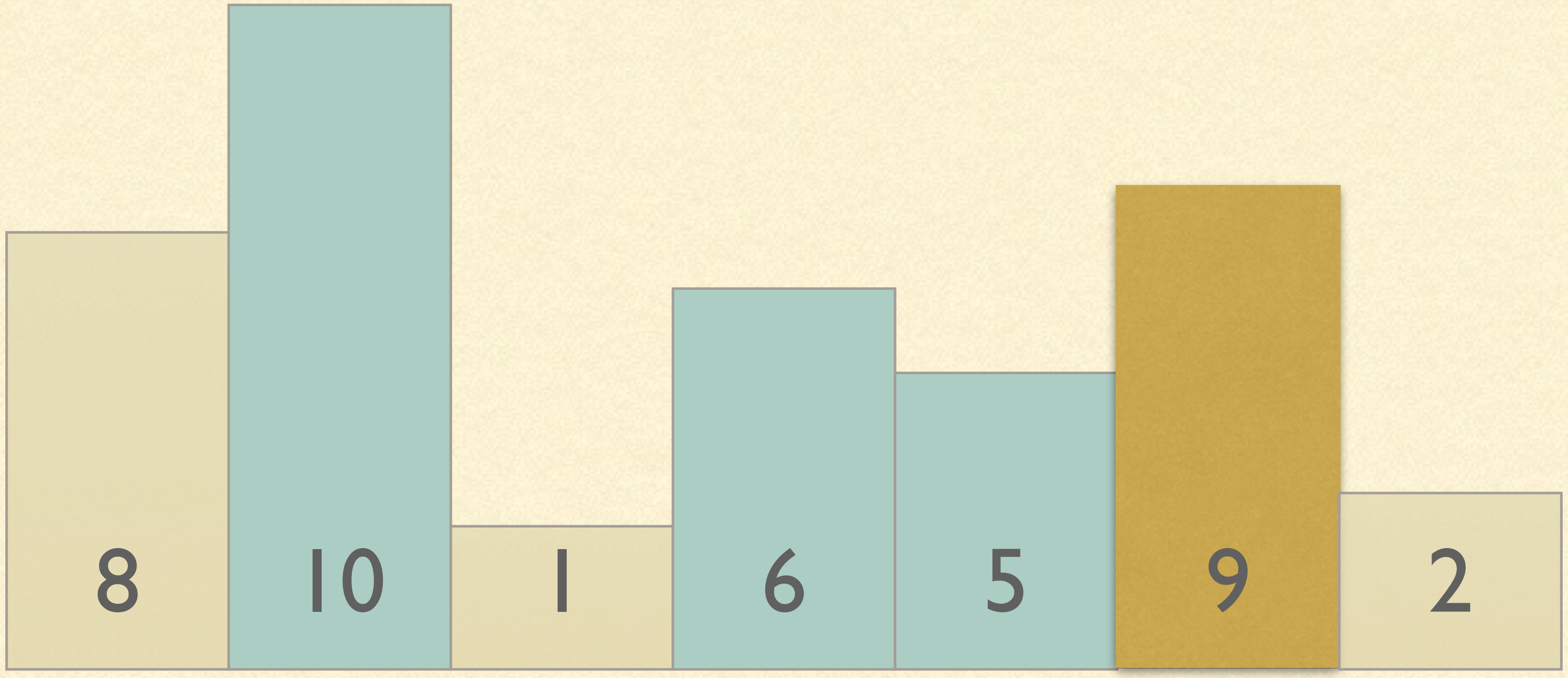


問題C　例



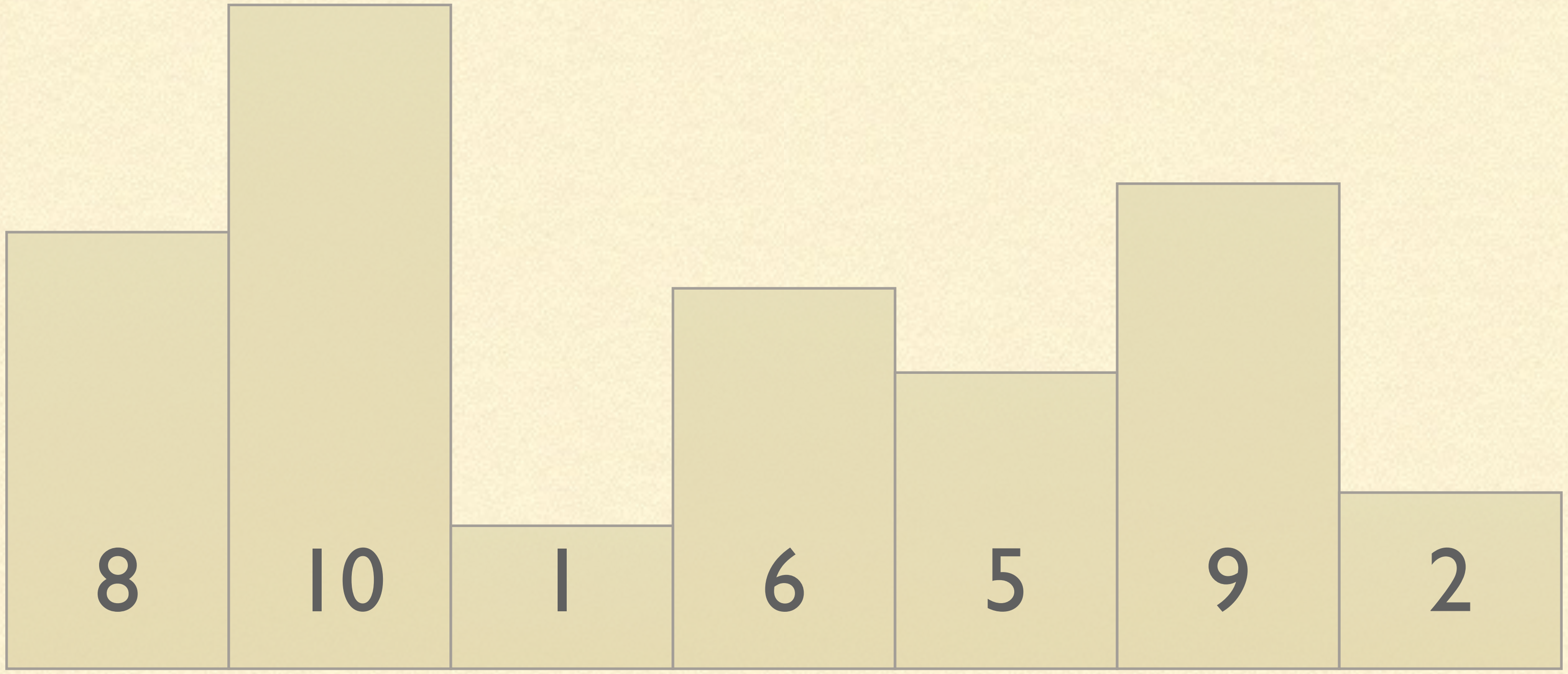
ここから見える人は、

問題C　例



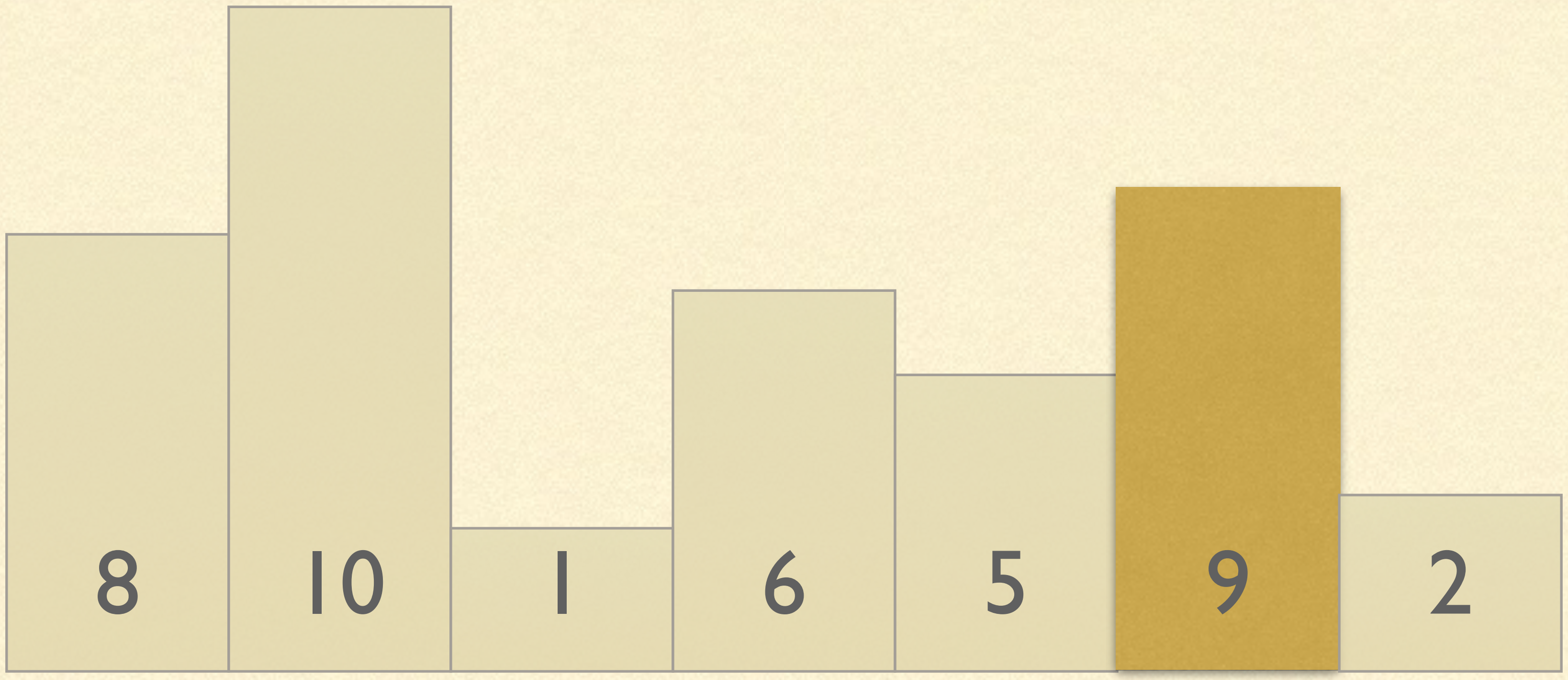
ここから見える人は、
3人

問題C　部分点解法（10点）



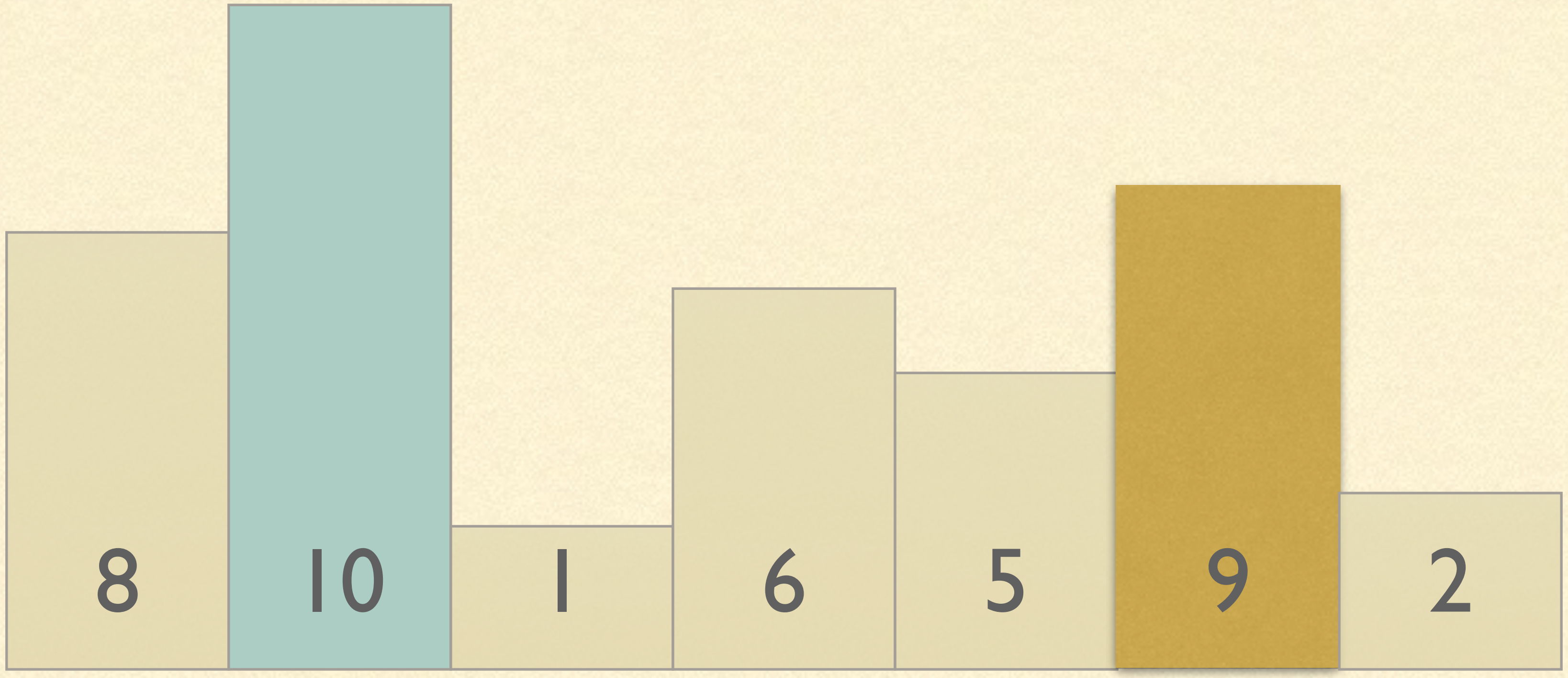
$n \leqq 100$

問題C　部分点解法（10点）



全ての人について、

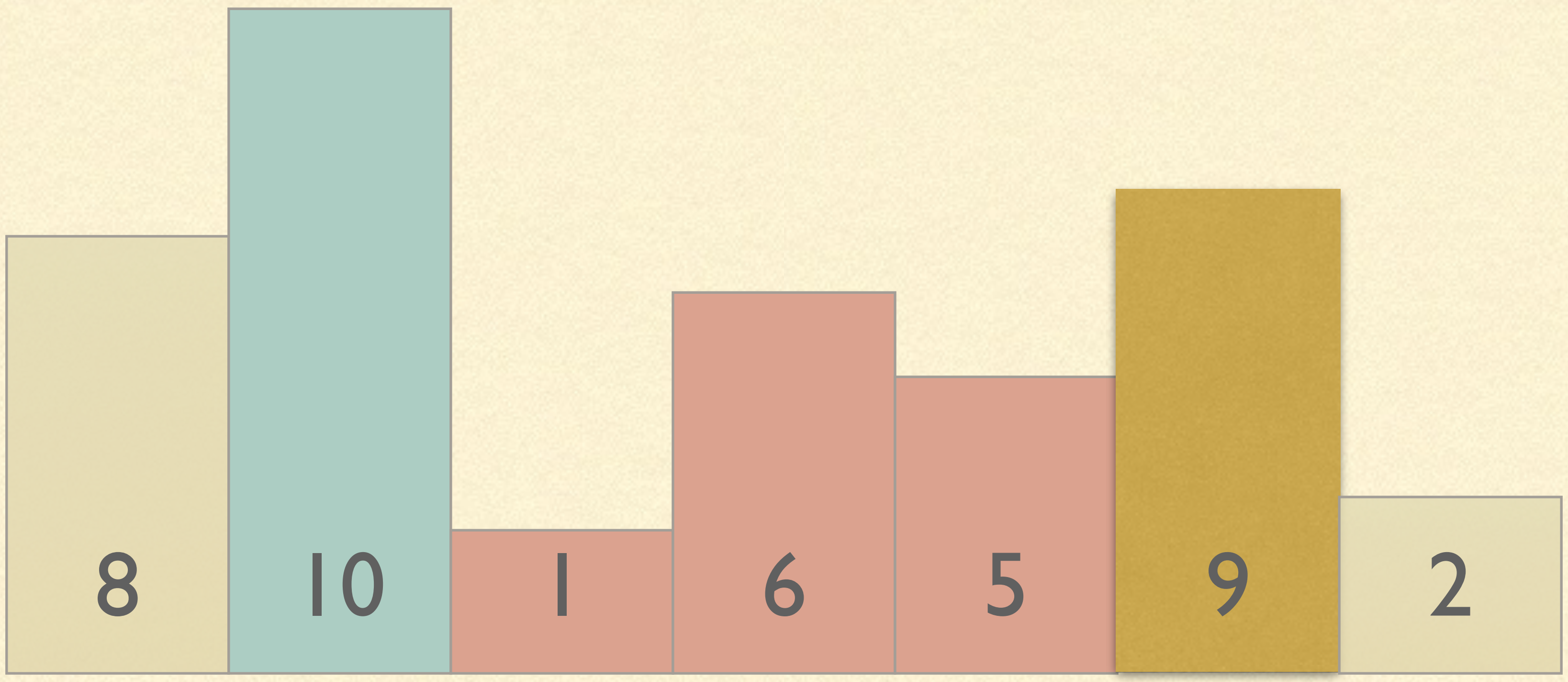
問題C　部分点解法（10点）



全ての人について、

別の全ての人に対し、

問題C　部分点解法（10点）

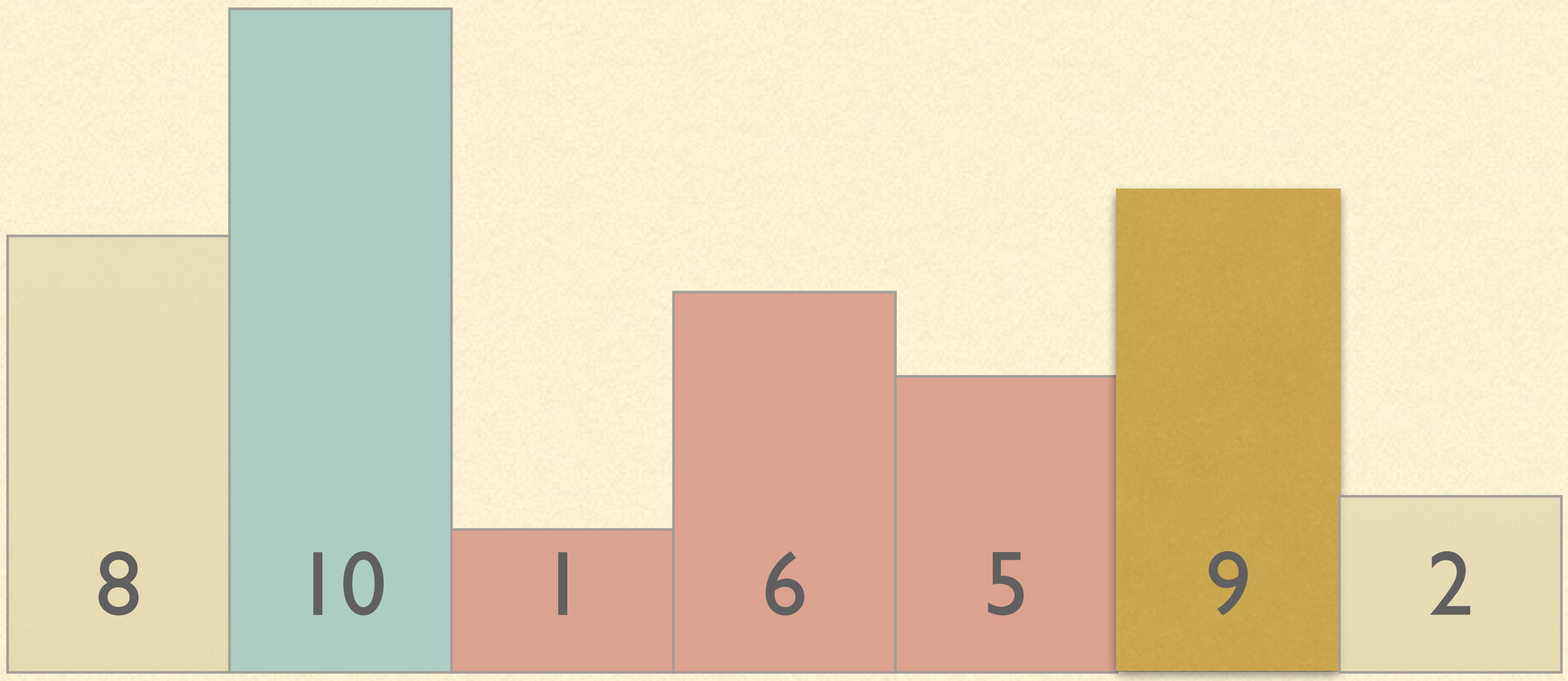


全ての人について、

別の全ての人に対し、

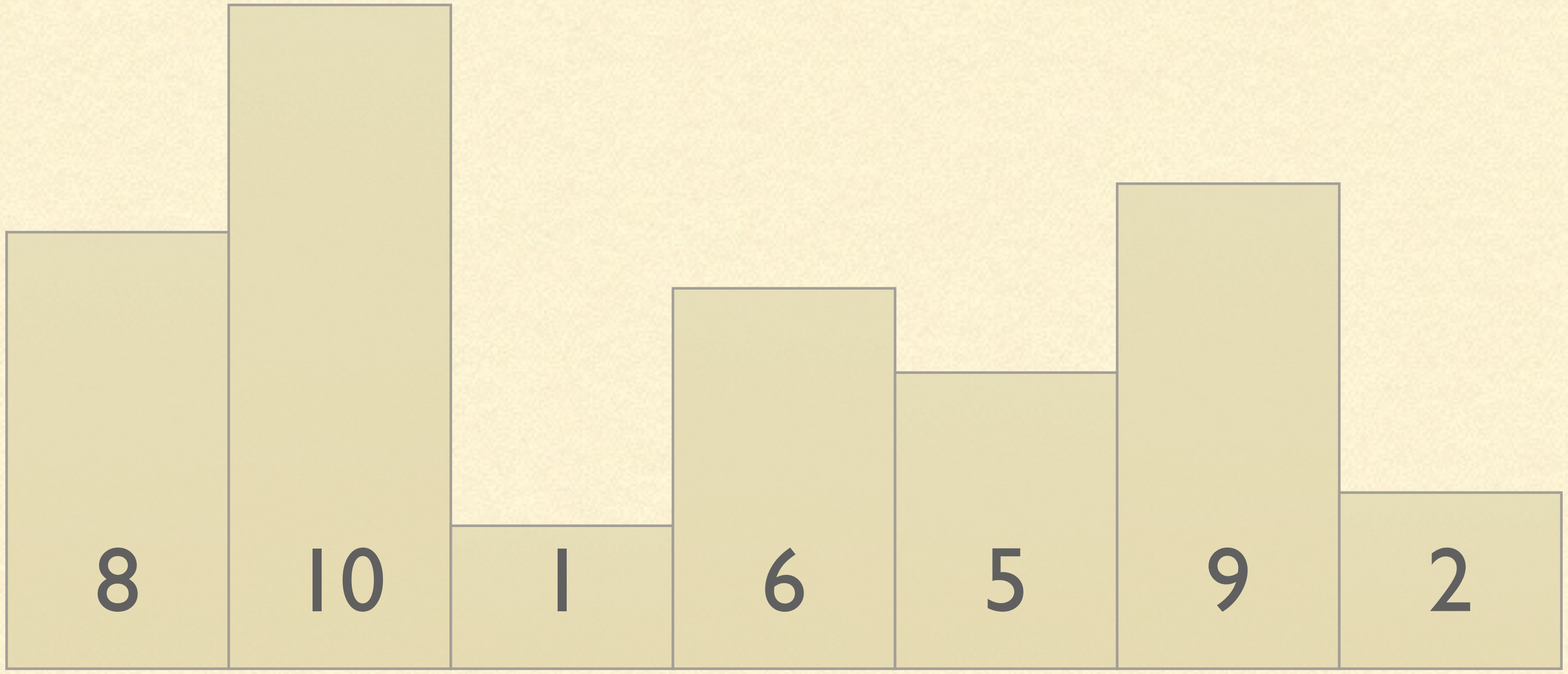
間に背の高い人がいないか調べる

問題C　部分点解法（10点）



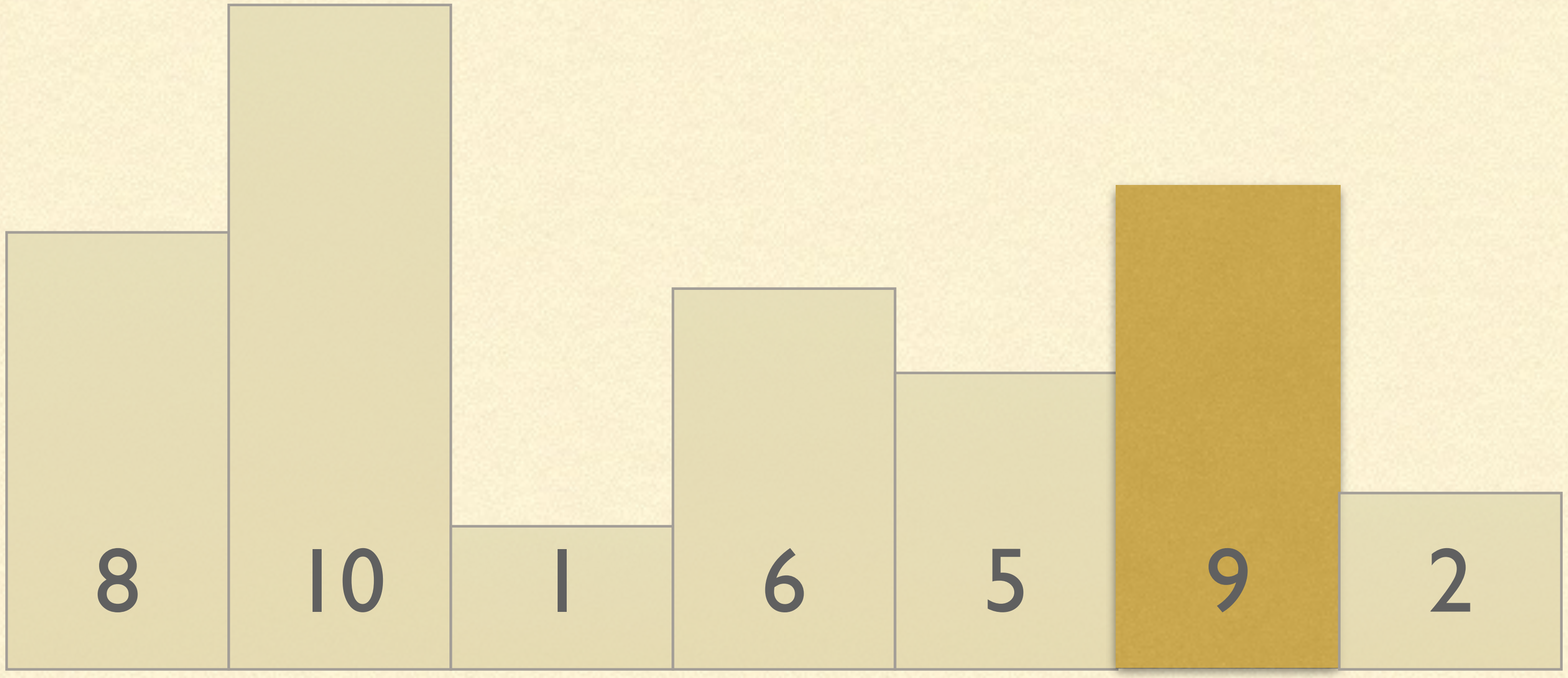
O(n^3)

問題C　部分点解法（40点）



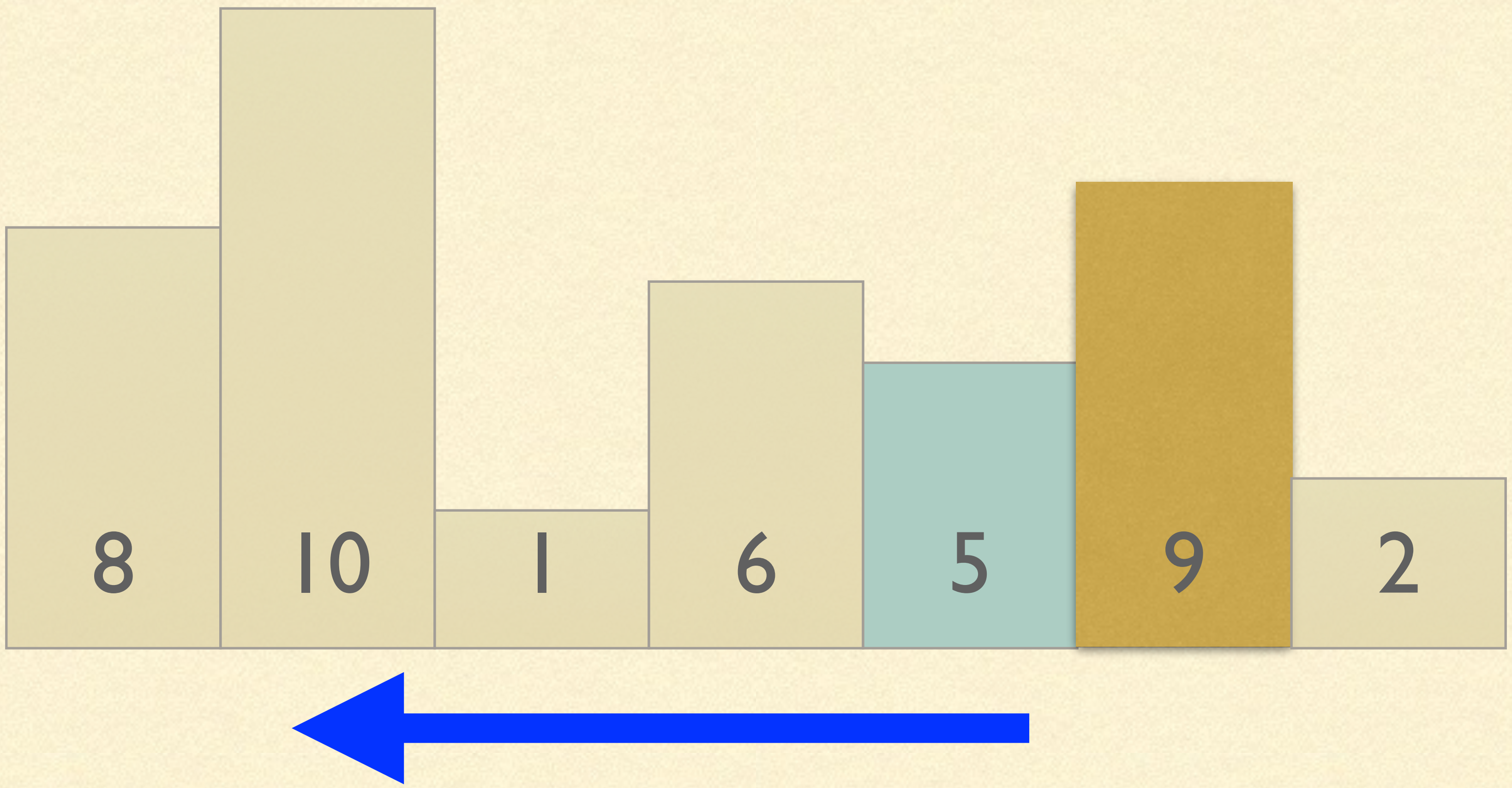
n ≦ 5000

問題C　部分点解法（40点）



調べたい人それぞれについて、

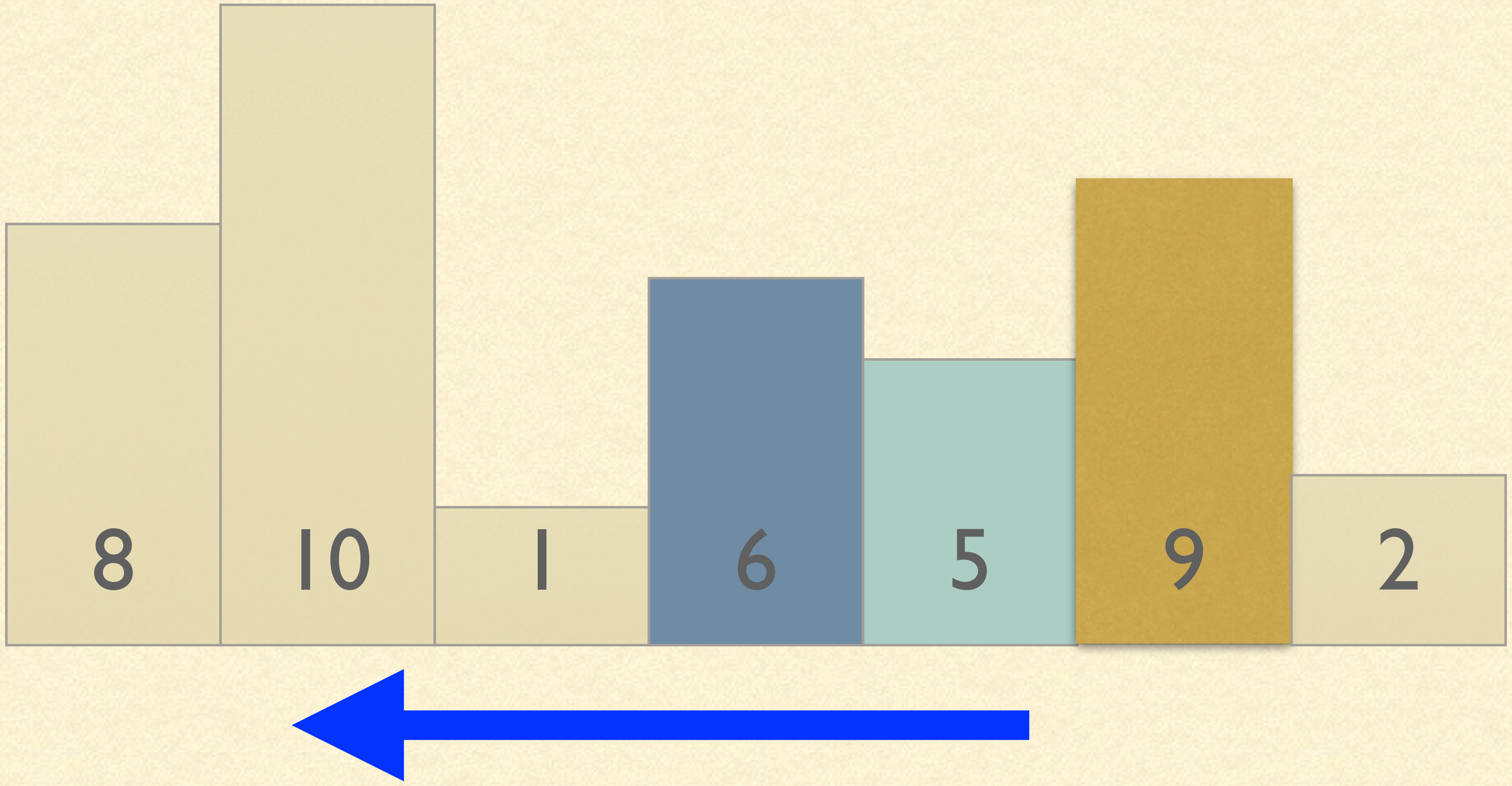
問題C　部分点解法（40点）



右から順に見ていき、

「それまでの最大値」を保持する

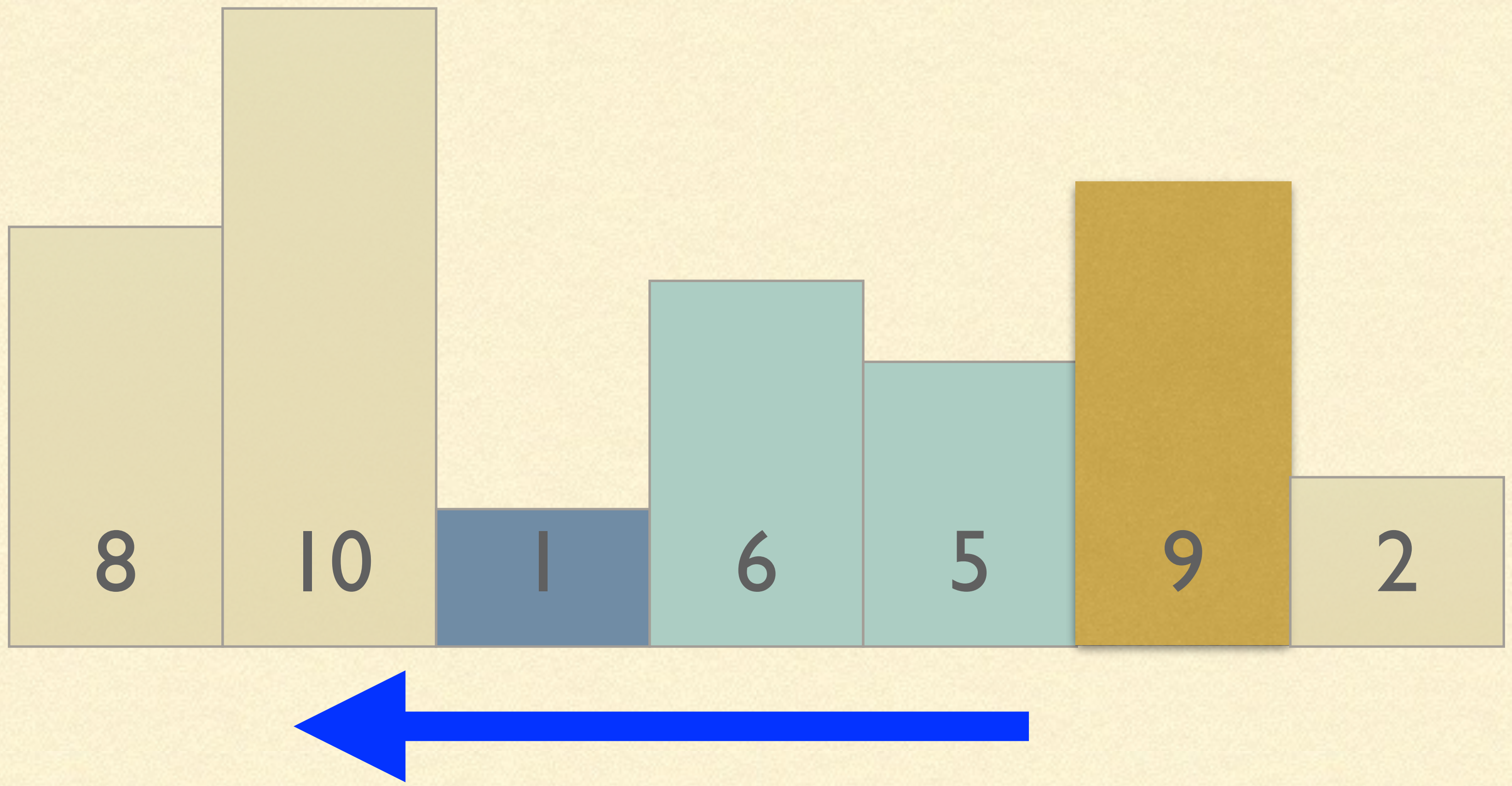
問題C　部分点解法（40点）



それまでの最大値より大きければ

カウント+1

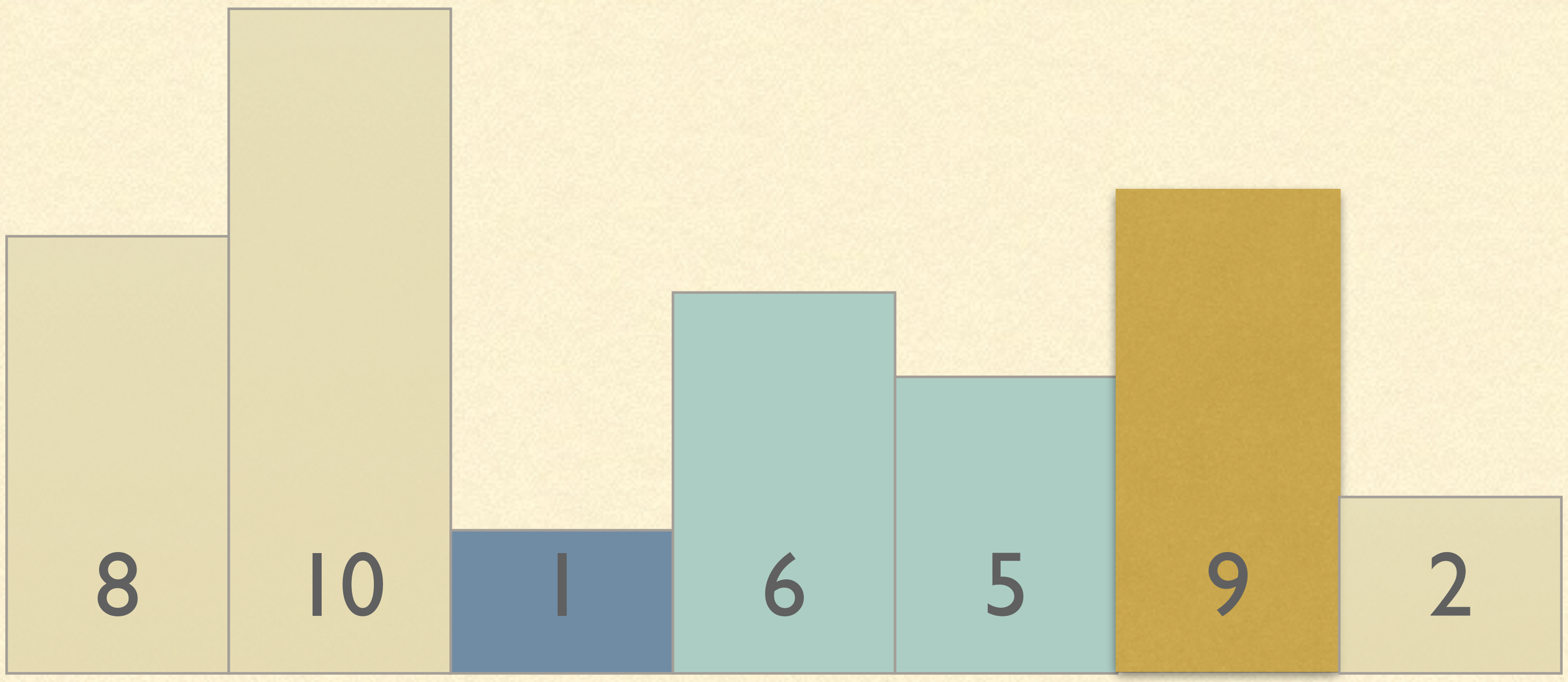
問題C　部分点解法（40点）



それまでの最大値より小さければ

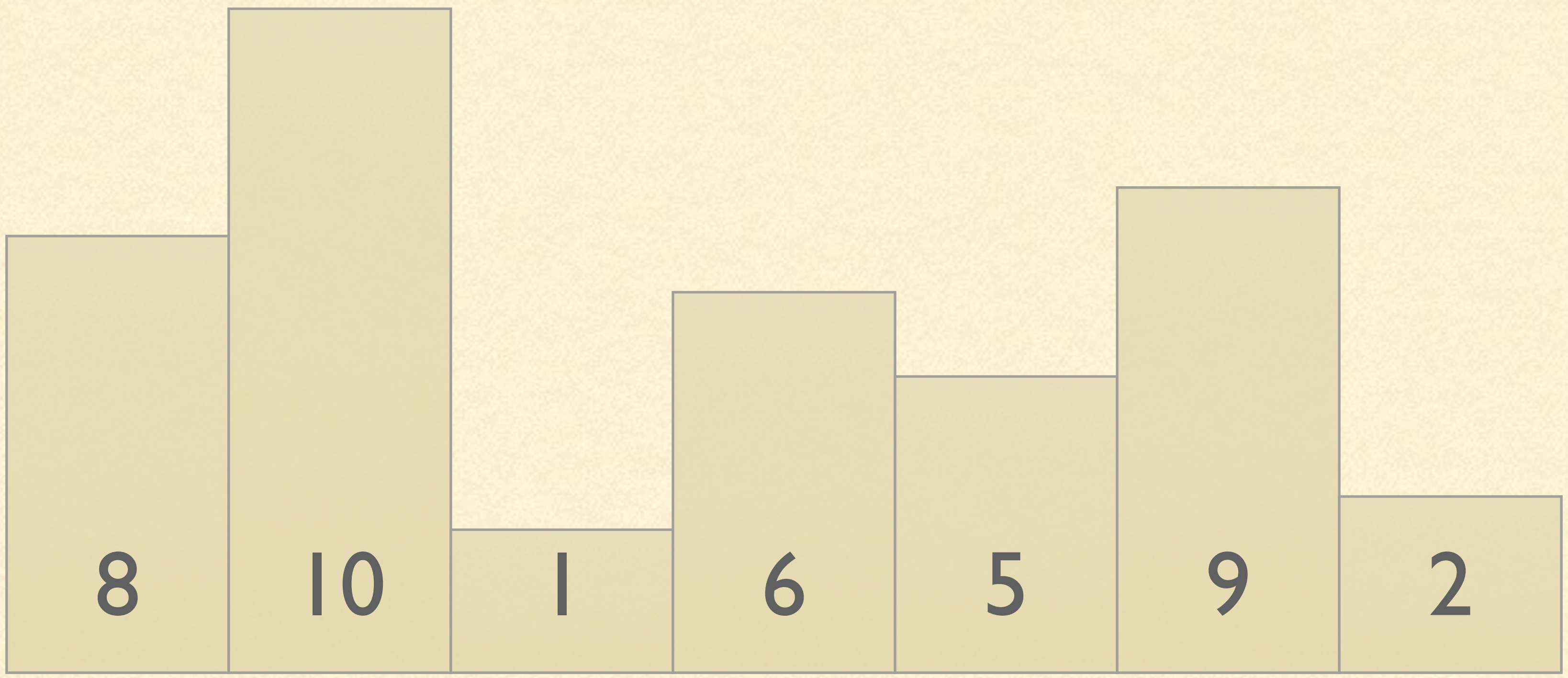
なにもしない

問題C　部分点解法（40点）



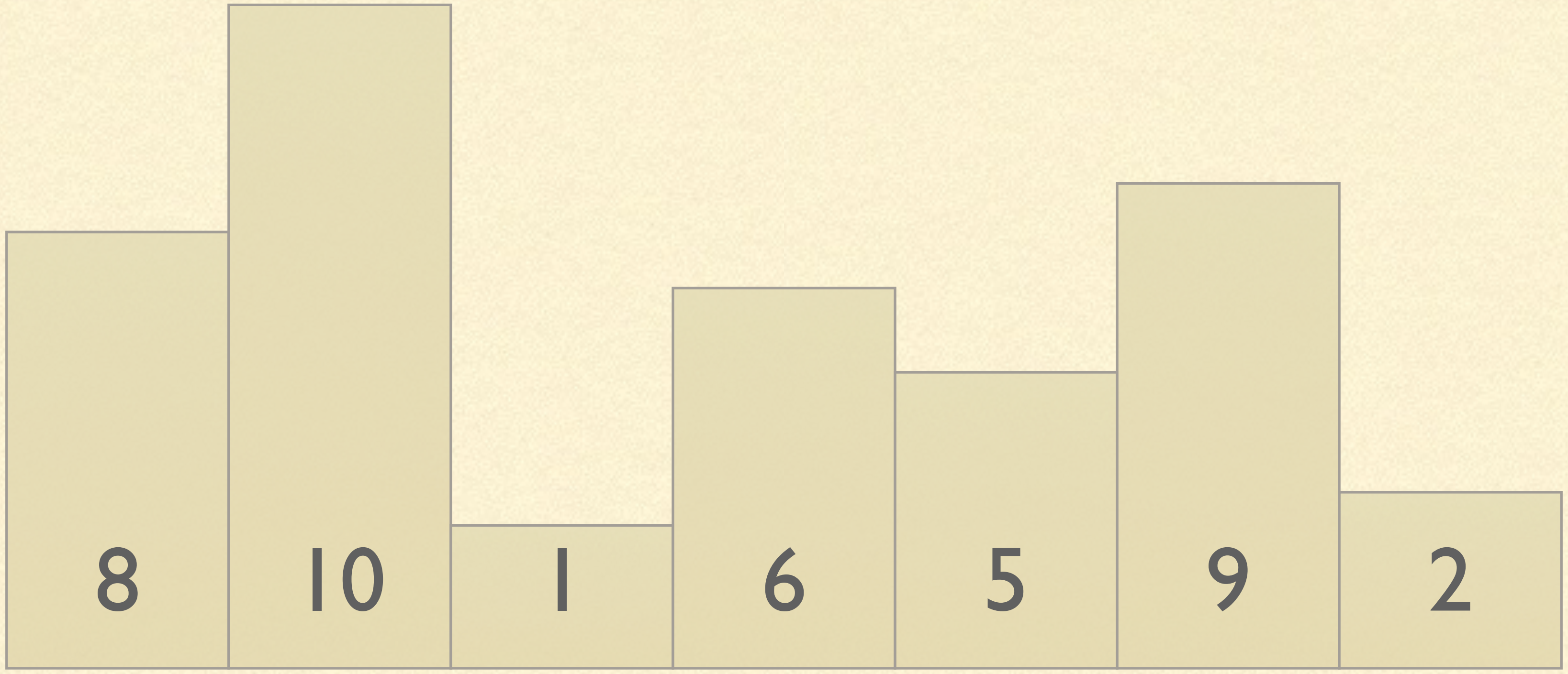
O(n^2)

問題C　満点解法（100点）



n ≦ 100000

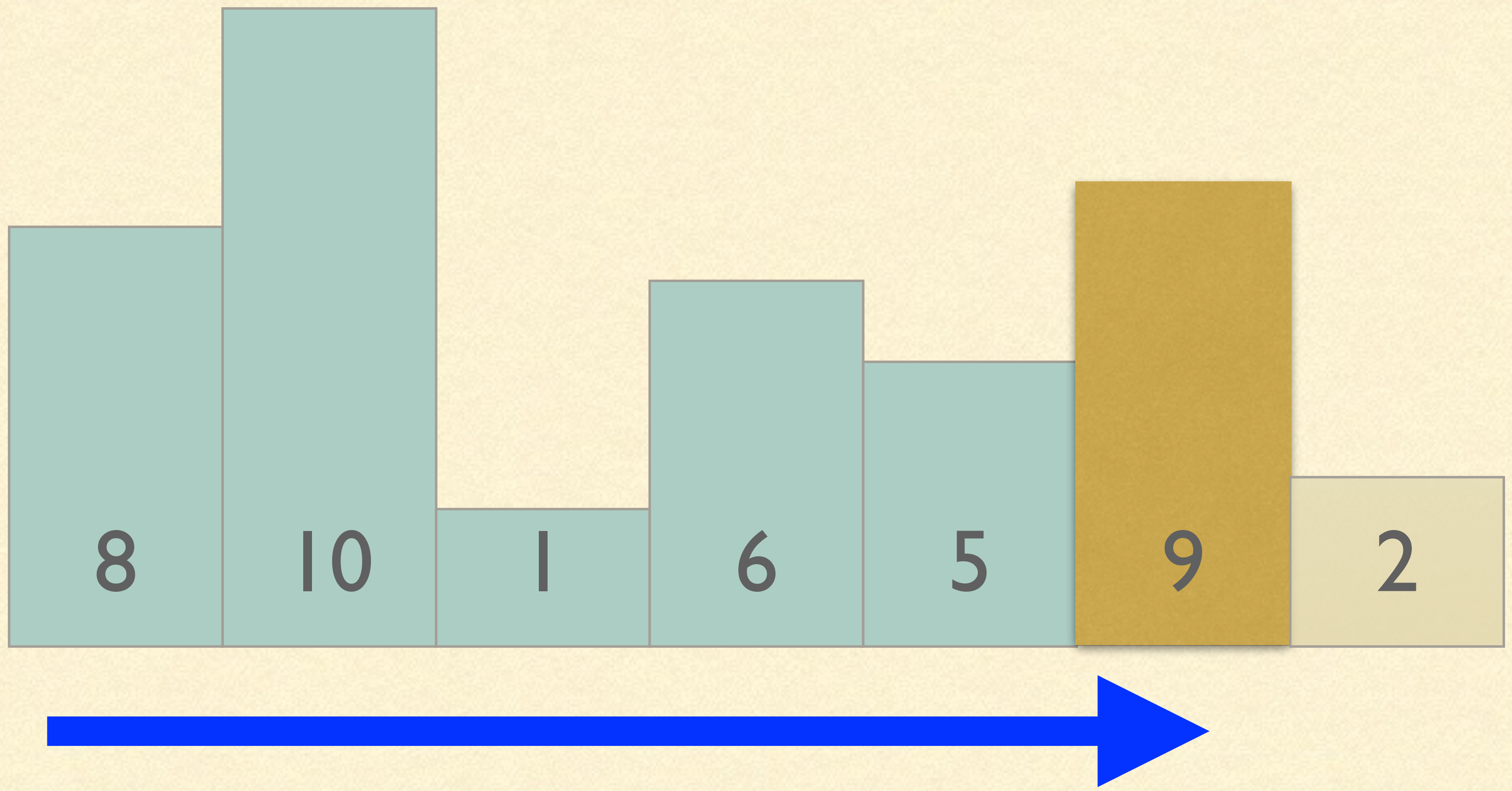
問題C　満点解法（100点）



n ≦ 100000

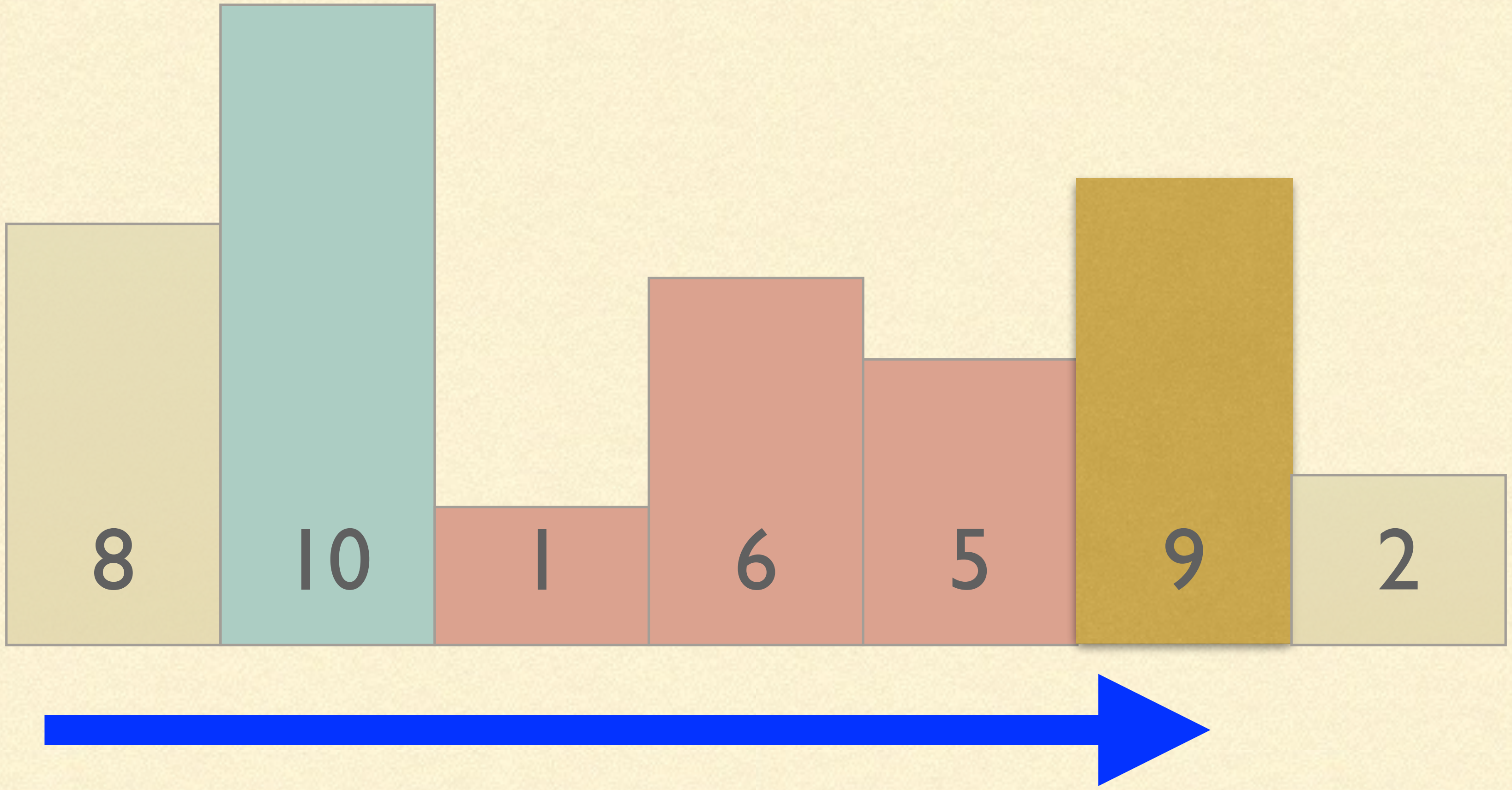
スタックを用いてO(n)で解く

問題C　満点解法（100点）



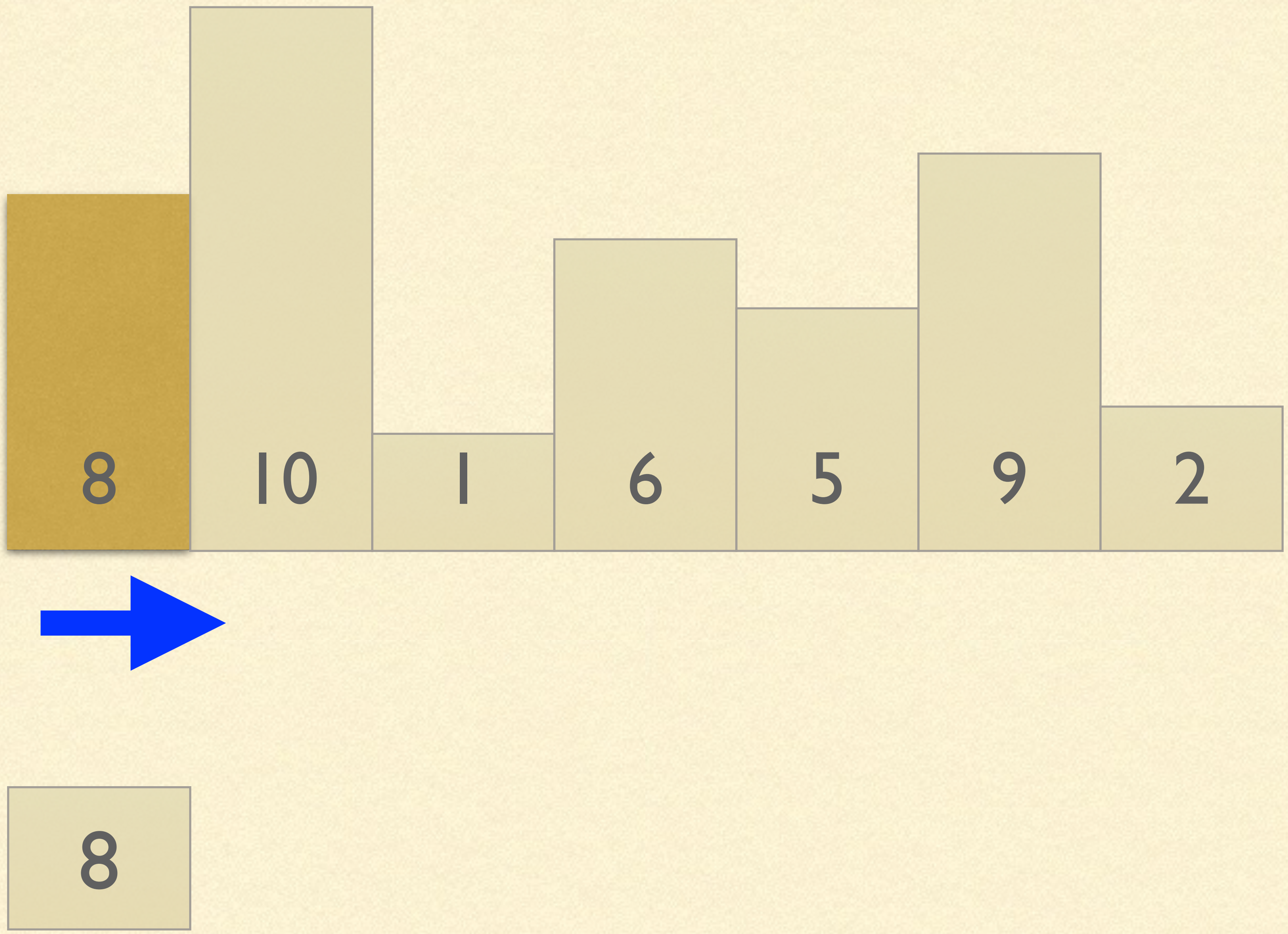
左から順に見て、

問題C　満点解法（100点）

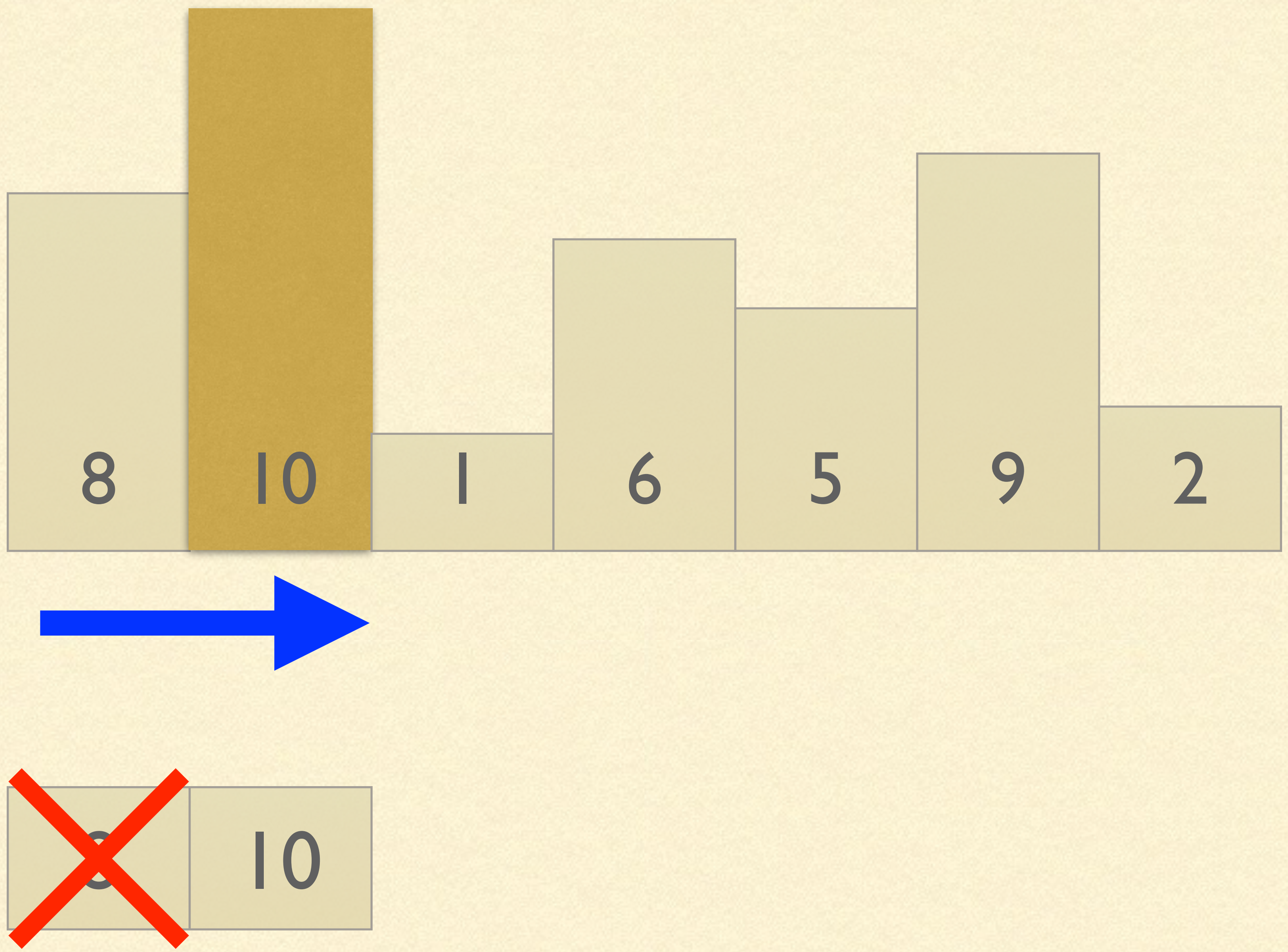


左から順に見て、
自分より背の低い情報はそれ以降捨てて良い

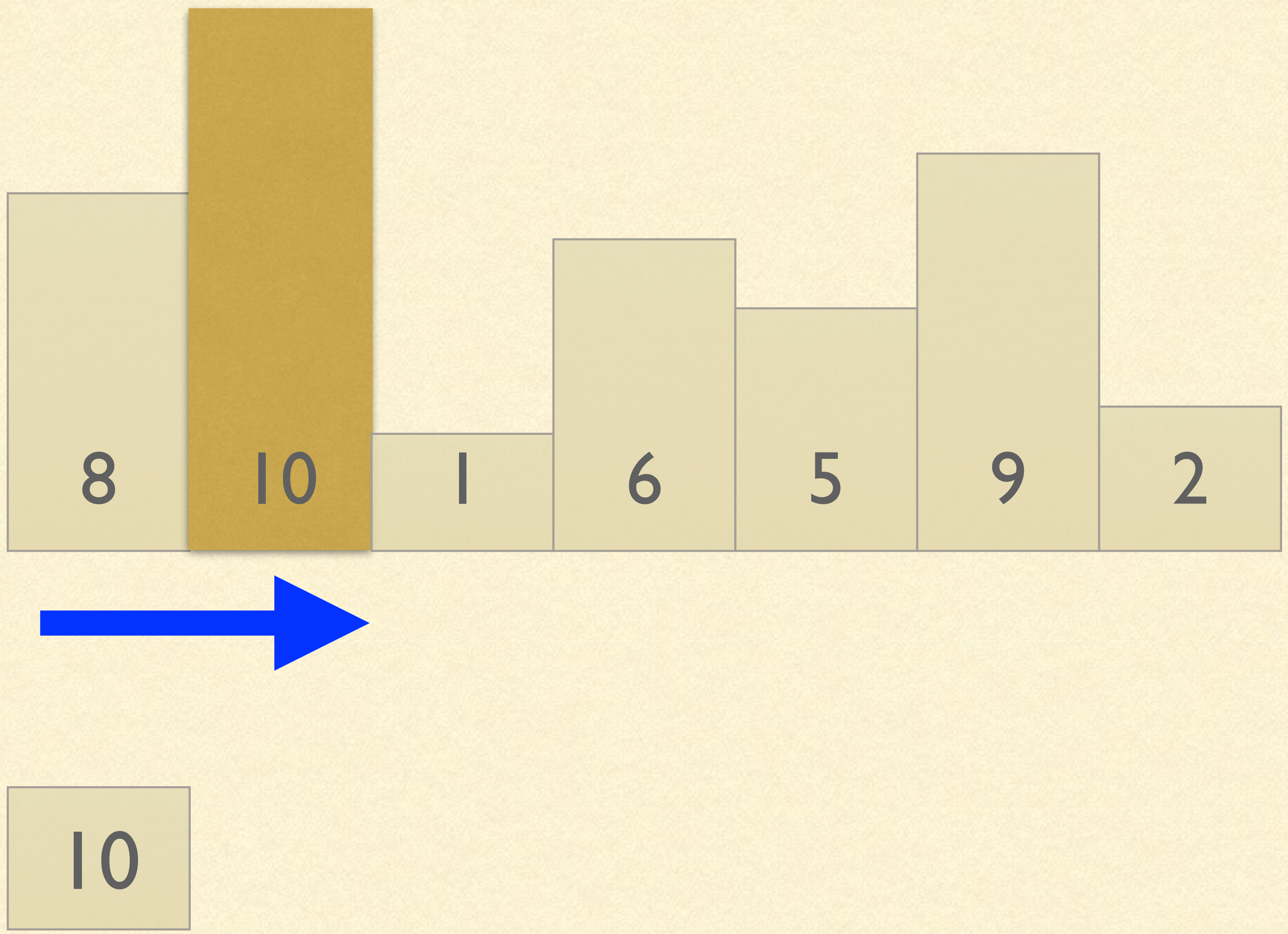
問題C　満点解法（100点）



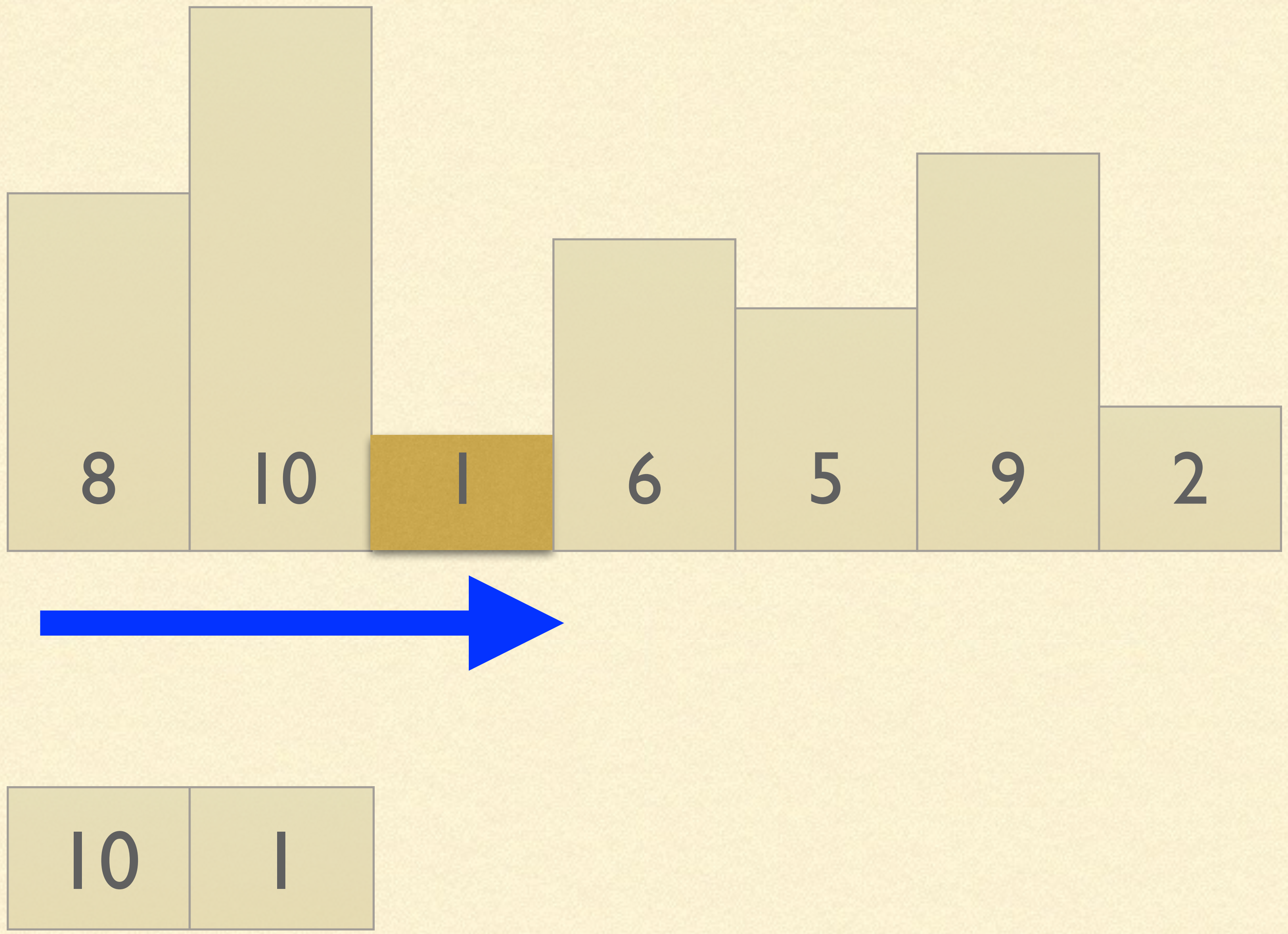
問題C　満点解法（100点）



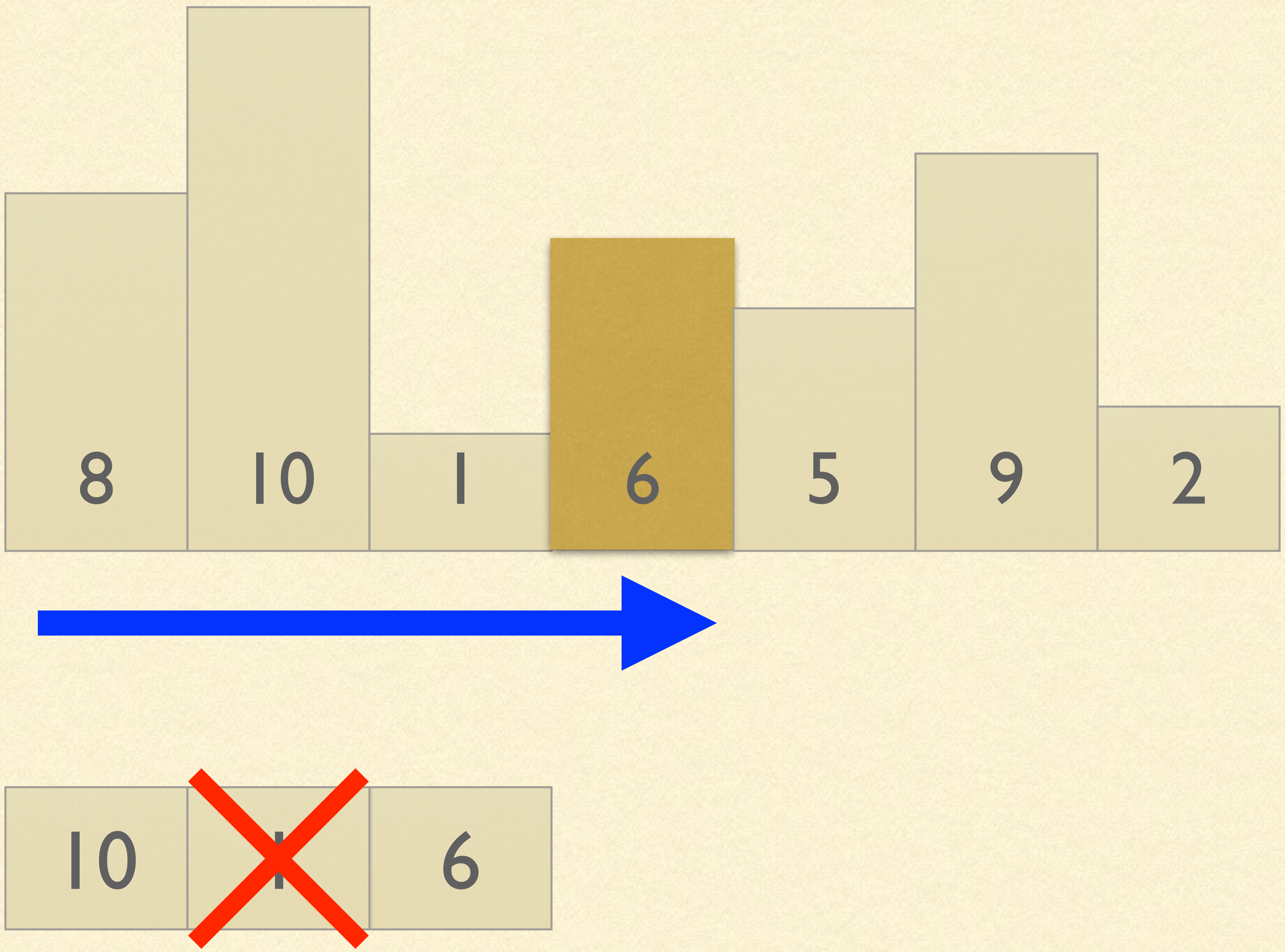
問題C　満点解法（100点）



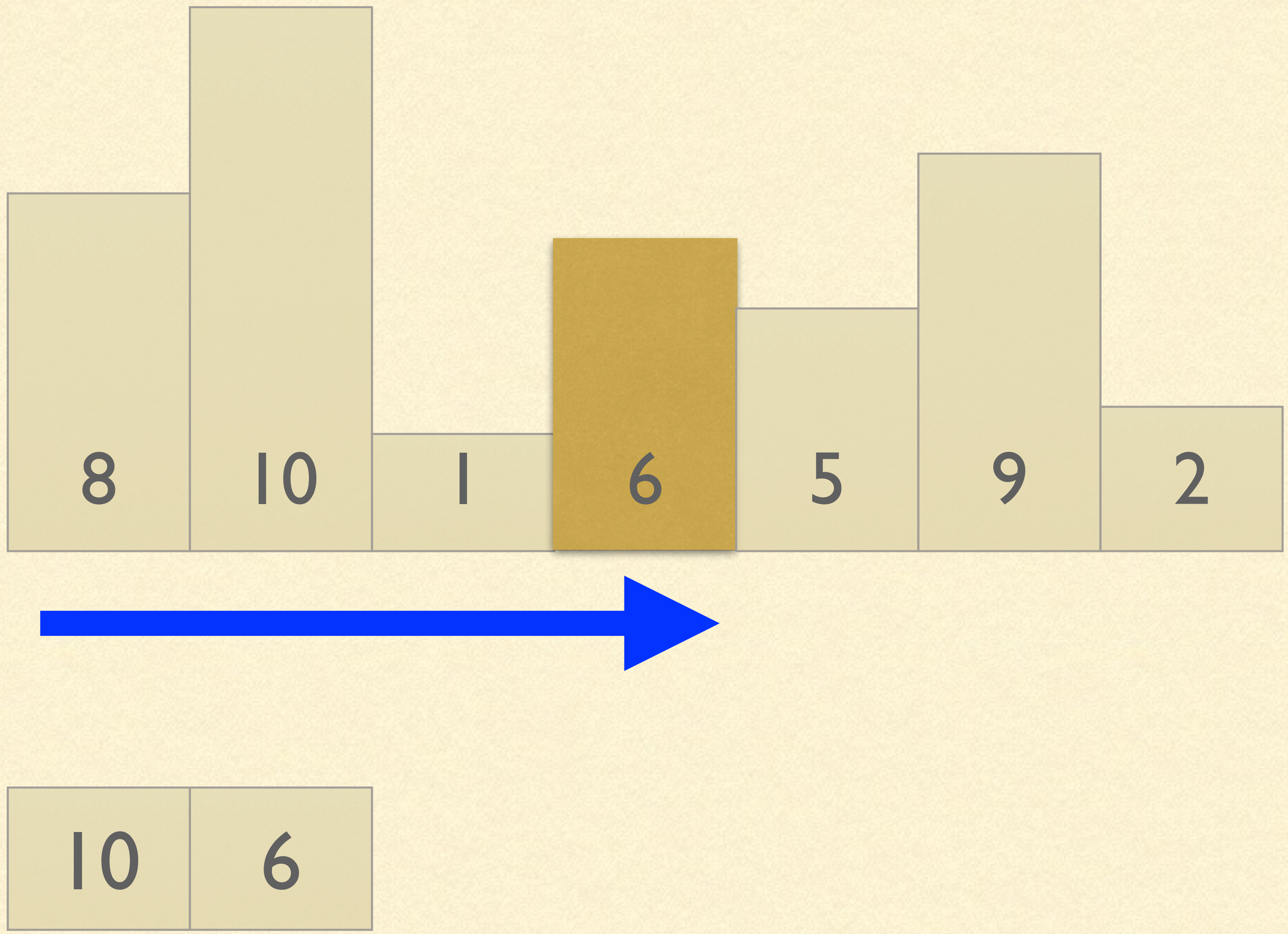
問題C　満点解法（100点）



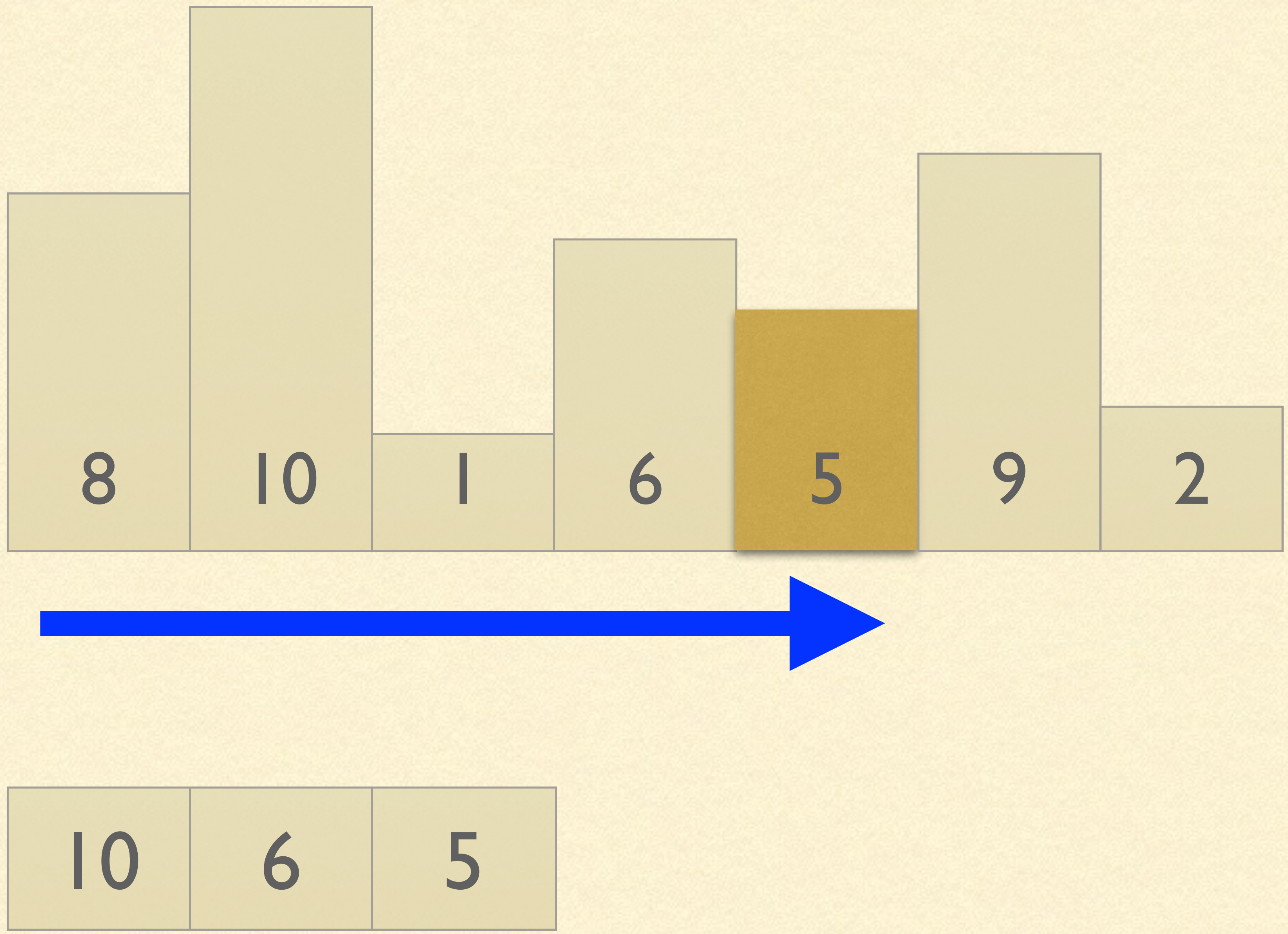
問題C　満点解法（100点）



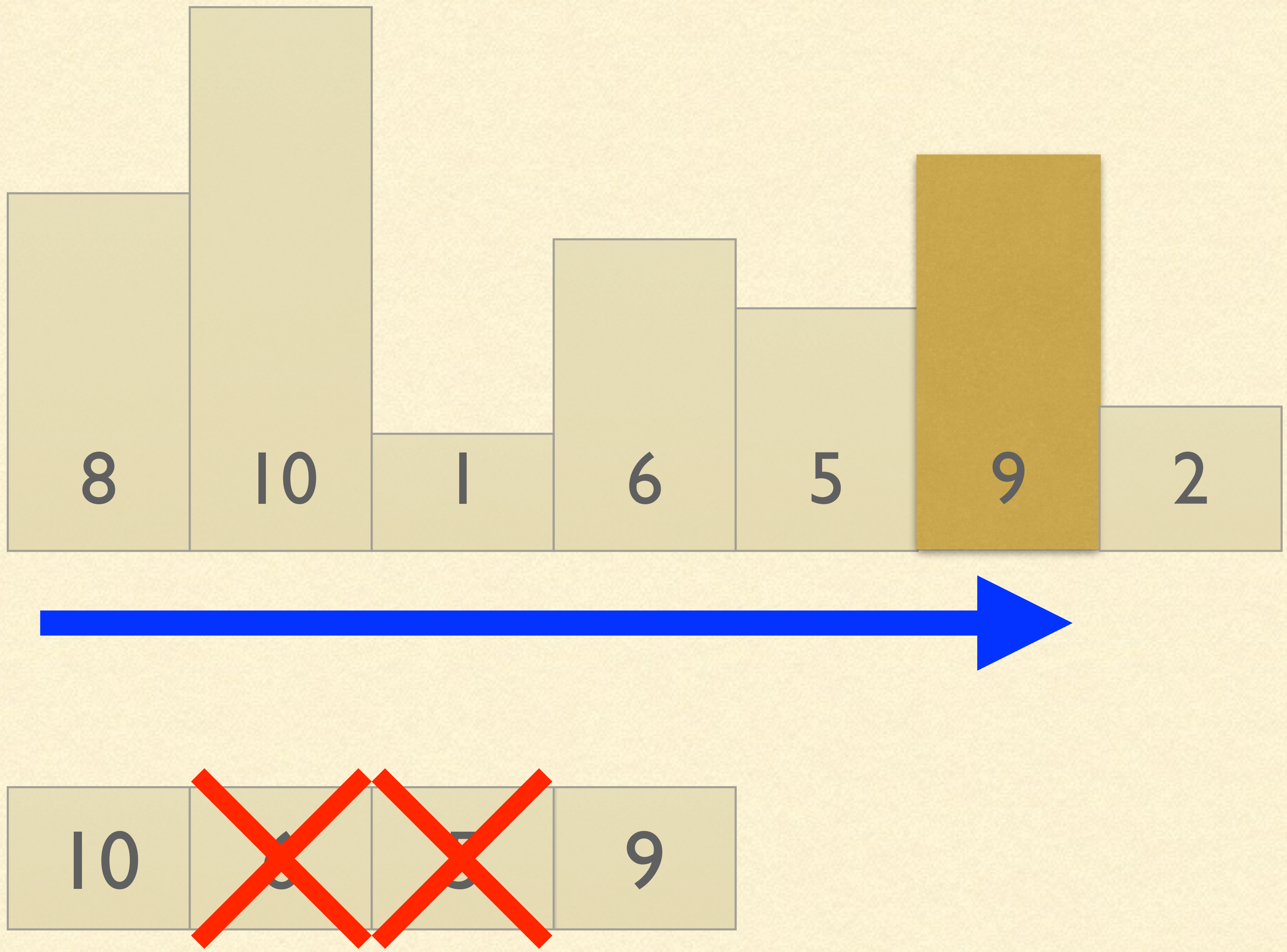
問題C　満点解法（100点）



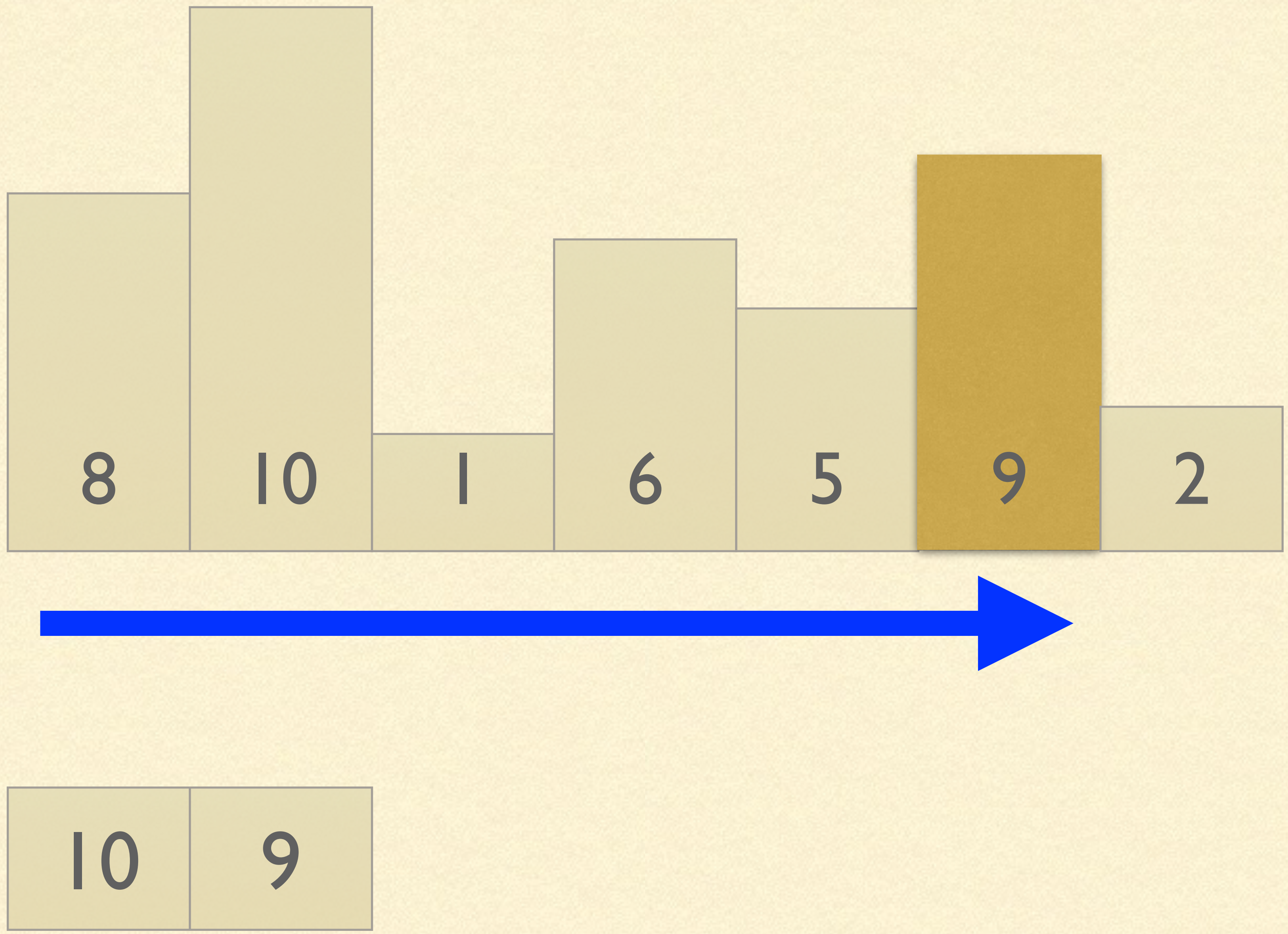
問題C　満点解法（100点）



問題C　満点解法（100点）



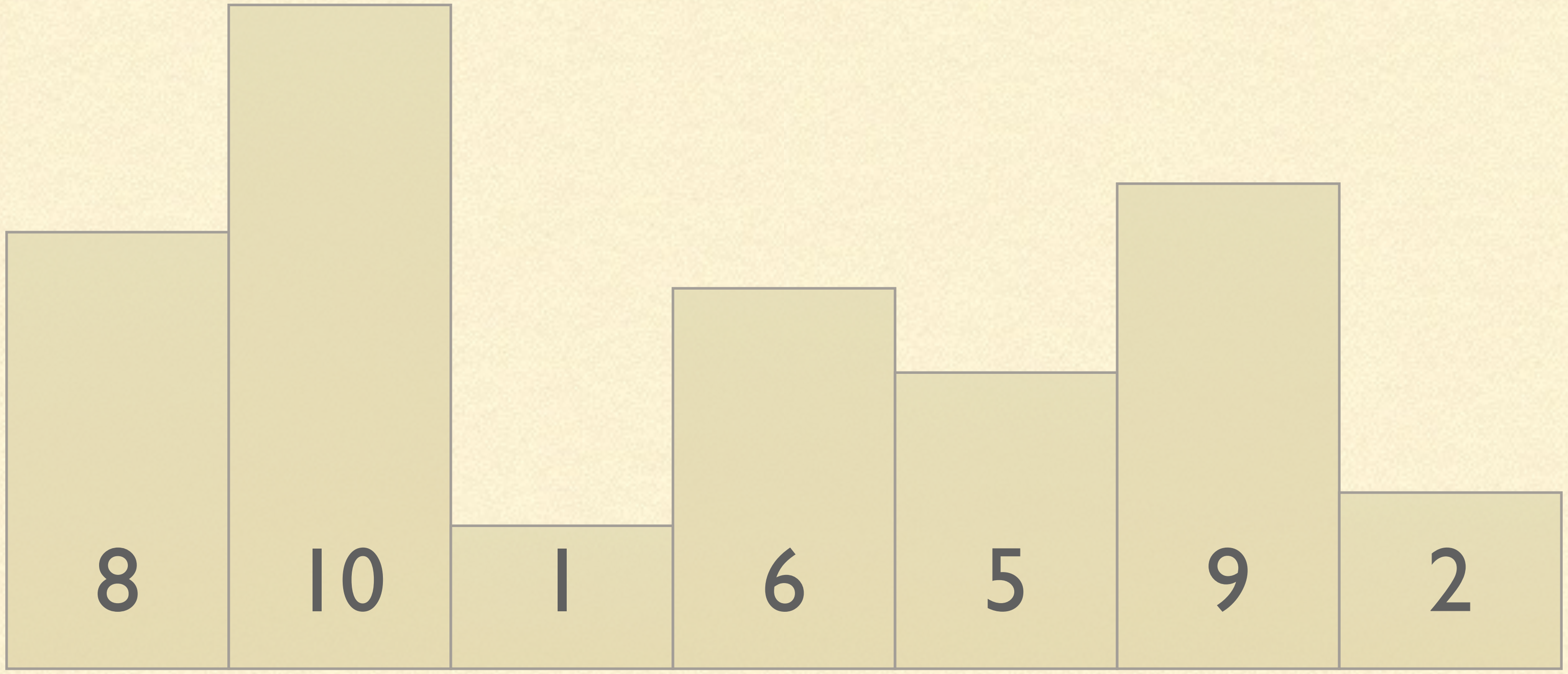
問題C　満点解法（100点）



問題C　満点解法（100点）

各要素に到達したときの
スタックのサイズが答え

問題C　満点解法（100点）



・「スタックに追加する」

・「スタックから取り除く」

いずれも高々n回　→　O(n)

問題D　ドーナツの箱詰め

問題D

- 1 ≦ n ≦ 200000 要素を、kグループに分ける
- 「各グループの（最大値 - 最小値）の和」の最小値は？

問題D　例

k = 3
{2, 12, 3, 13, 7, 17, 1}

問題D　例

k = 3
{2, 12, 3, 13, 7, 17, 1}

{1, 2, 3, 7}	{12, 13}	{17}

6+1+0 = 7 が最小値

問題D　例

k = 3
{2, 12, 3, 13, 7, 17, 1}

どんな入力であっても、

問題D　例

k = 3
{1, 2, 3, 7, 12, 13, 17}

どんな入力であっても、
ソートしておき、

問題D　例

k = 3

{1, 2, 3, 7, 12, 13, 17}

どんな入力であっても、
ソートしておき、
重ならない区間でグループ分けする
ことが最小値の必要条件

問題D　部分点解法（15点）

{1, 2, 3, 7, 12, 13, 17}

k = 2

問題D　部分点解法（15点）

$$\{\underbrace{1,\ 2,\ 3}_{},\ \underbrace{7,\ 12,\ 13,\ 17}_{}\}$$

k = 2

ソート後、グループ分けの候補を

n-1通り試せば良い

問題D　部分点解法（40点）

{1, 2, 3, 7, 12, 13, 17}

$1 \leqq k \leqq n$

問題D　部分点解法（40点）

{1,　2,　3,　7,　12,　13,　17}

1　1　4　5　1　4

各要素の差からなる(n-1)要素を準備

問題D　部分点解法（40点）

{1,　2,　3,　7,　12,　13,　17}

k = 3　　1　1　4　5　1　4

各要素の差からなる(n-1)要素を準備

グループ分けした時、差の要素のうち

(k-1)要素は含まれない

問題D　部分点解法（40点）

{1,　2,　3,　7,　12,　13,　17}

k = 3　1　1　4　5　1　4

小さい方から(n-1)-(k-1) = n-k個

を選べば良い

問題D　部分点解法（45点）

{1,　2,　3,　7,　12,　13,　17}

1　1　4　5　1　4

k = 2

問題D　部分点解法（45点）

{1,　2,　3,　7,　12,　~~13~~,　17}

1　1　4　5　5

“差の要素”が減っていく

k = 2

問題D　部分点解法（45点）

{1,　2,　3,　7,　12,　~~13~~,　17}

1　1　4　5　　5

もとの要素はsetや隣接リストなどで持っておき、要素がなくなった時に消える“差”と追加される“差”を計算する。

問題D　部分点解法（45点）

{1,　2,　3,　7,　12,　~~13~~,　17}

1　1　4　5　　5

“差”の要素はmultisetなどで持っておき、

“差の合計”から“差の最大値”を引けば良い。

問題D　満点解法（100点）

{1,　2,　3,　7,　12,　~~13~~,　17}

1　1　4　5　　5

“差の要素”が減っていく

$1 \leqq k \leqq n$

問題D　満点解法（100点）

{1,　2,　3,　7,　12,　~~13~~,　17}

1　1　4　5　　5

・“差の要素”に数Xを追加する／取り除く

・“差の要素”の小さいn-k個の和を求める

という操作を高速に行う

問題D　満点解法（100点）

{1,　2,　3,　7,　12,　~~13~~,　17}

1　1　4　5　　5

・“差の要素”に数Xを追加する／取り除く

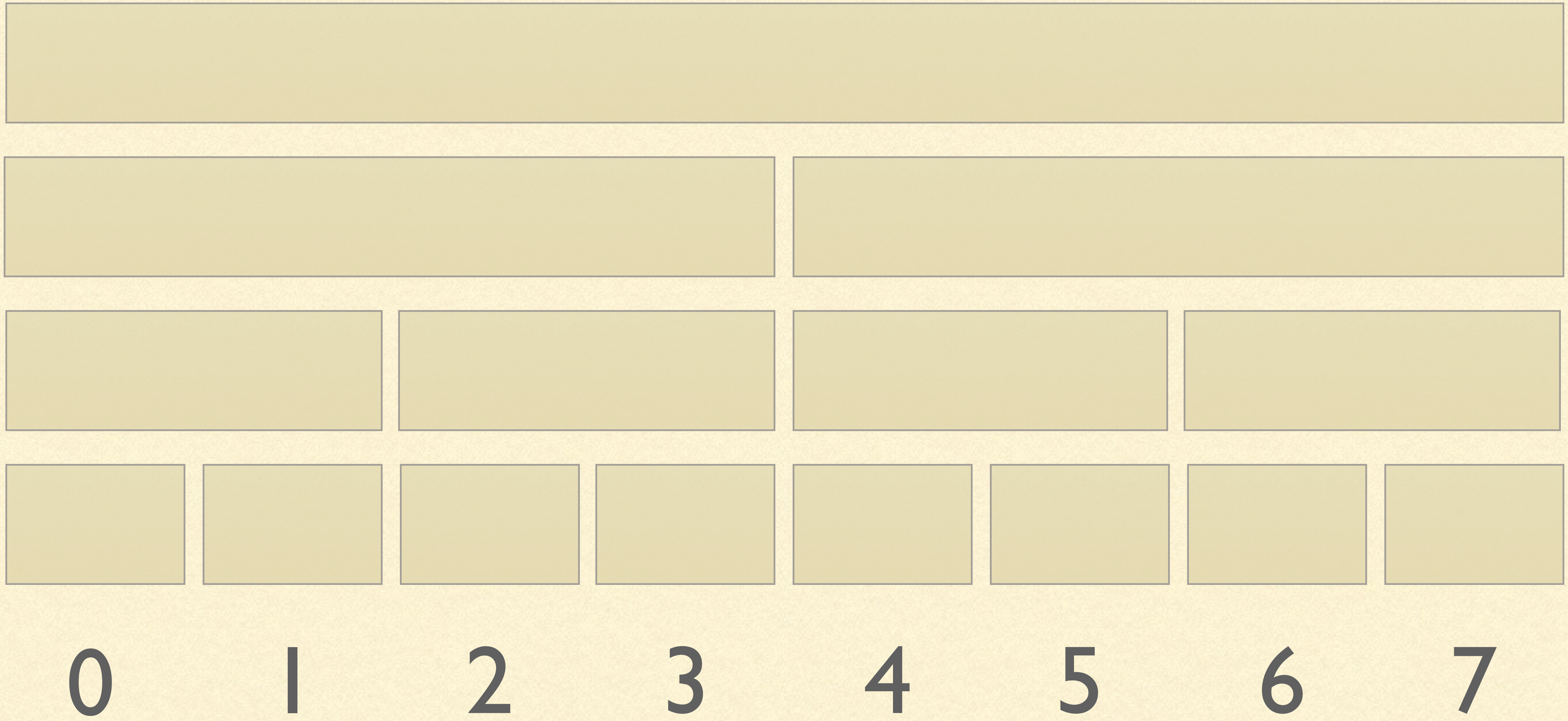
・“差の要素”の小さいn-k個の和を求める

という操作を高速に行う

Segment Tree

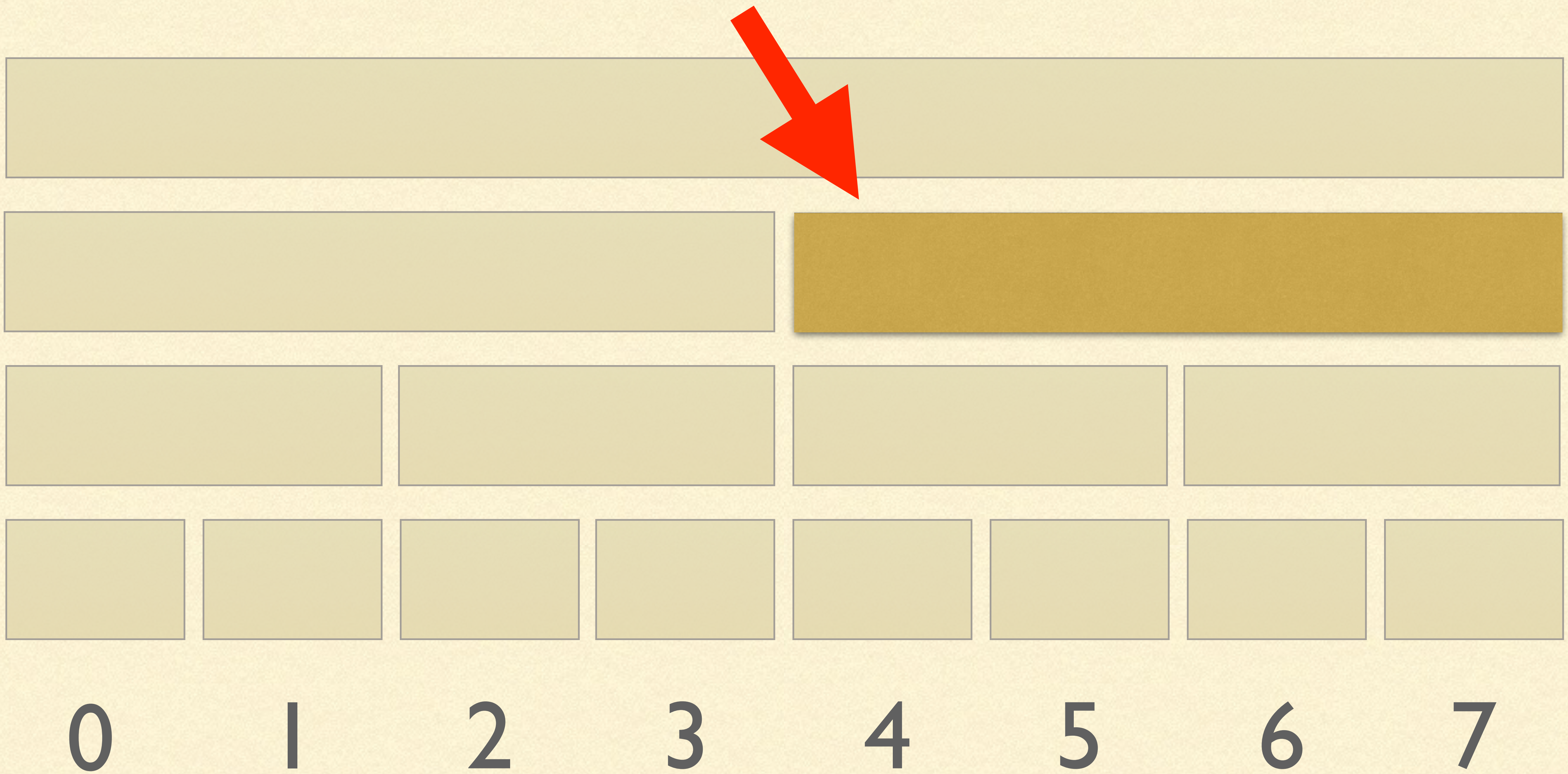
各ノードは、“差の要素”の

「個数」と「和」の情報をもつ。



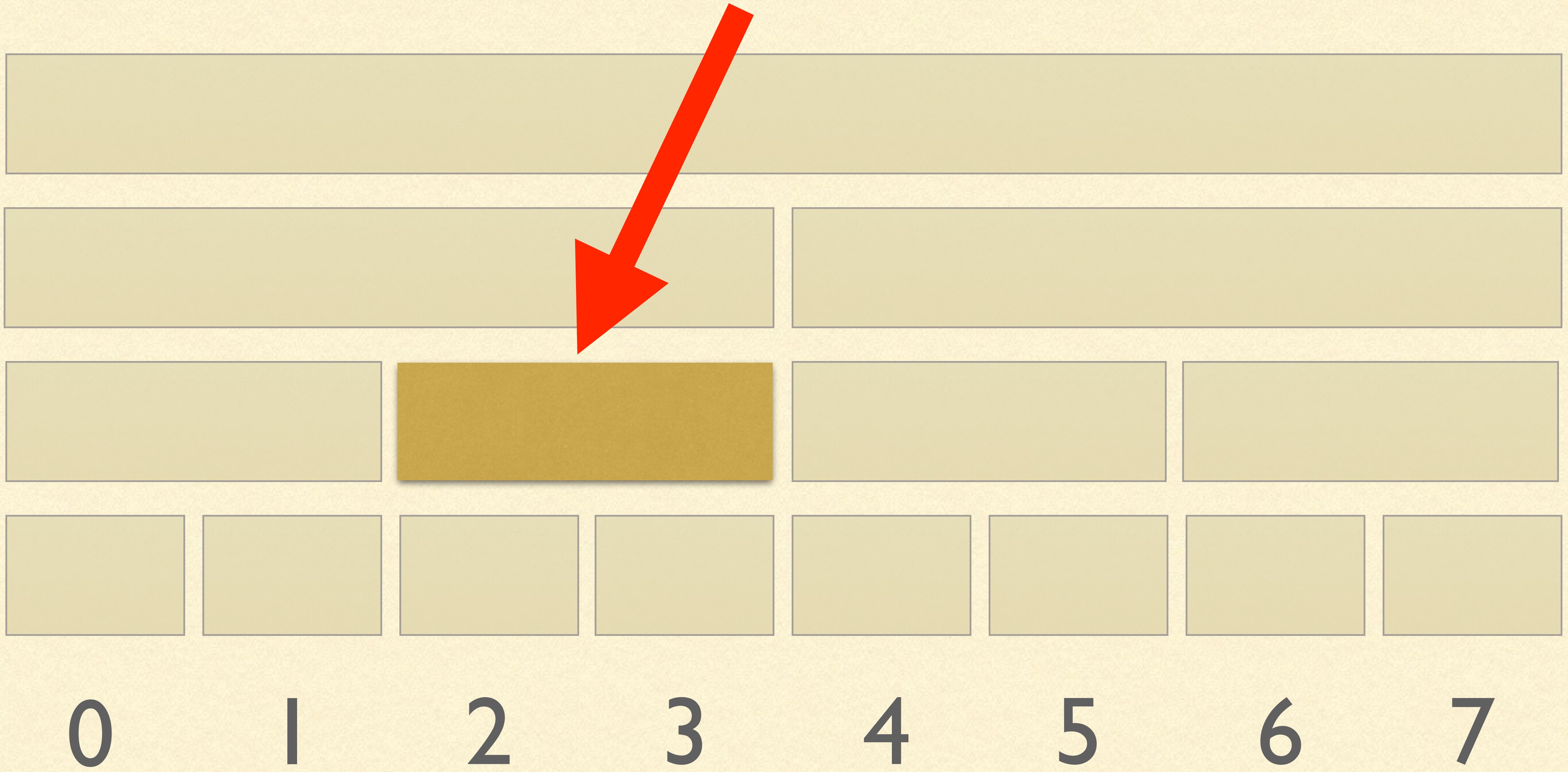
“差の要素”の中に、4〜7は、

現在いくつ持っていて、その合計はいくつか？

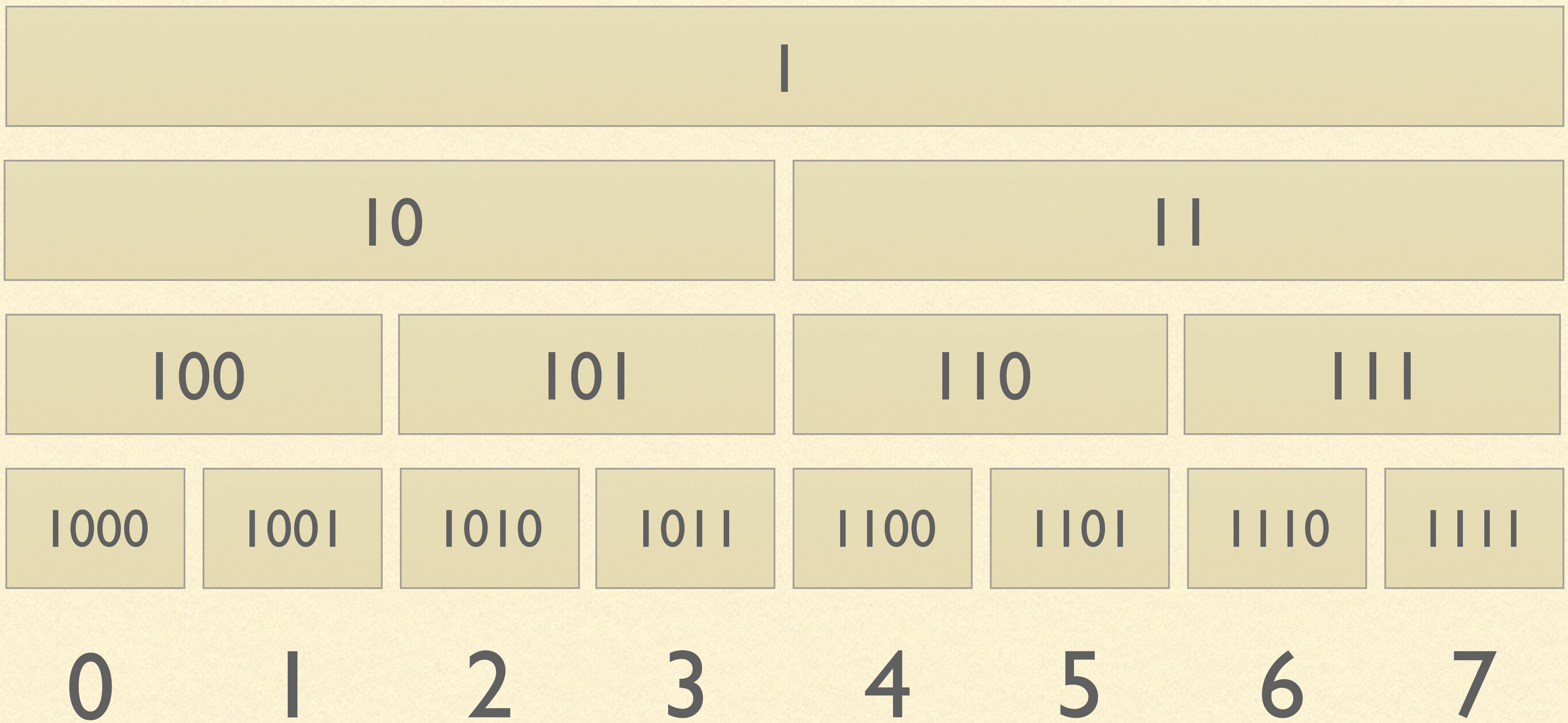


2・3は、現在いくつ持っていて、

その合計はいくつか？



ノード番号（２進）

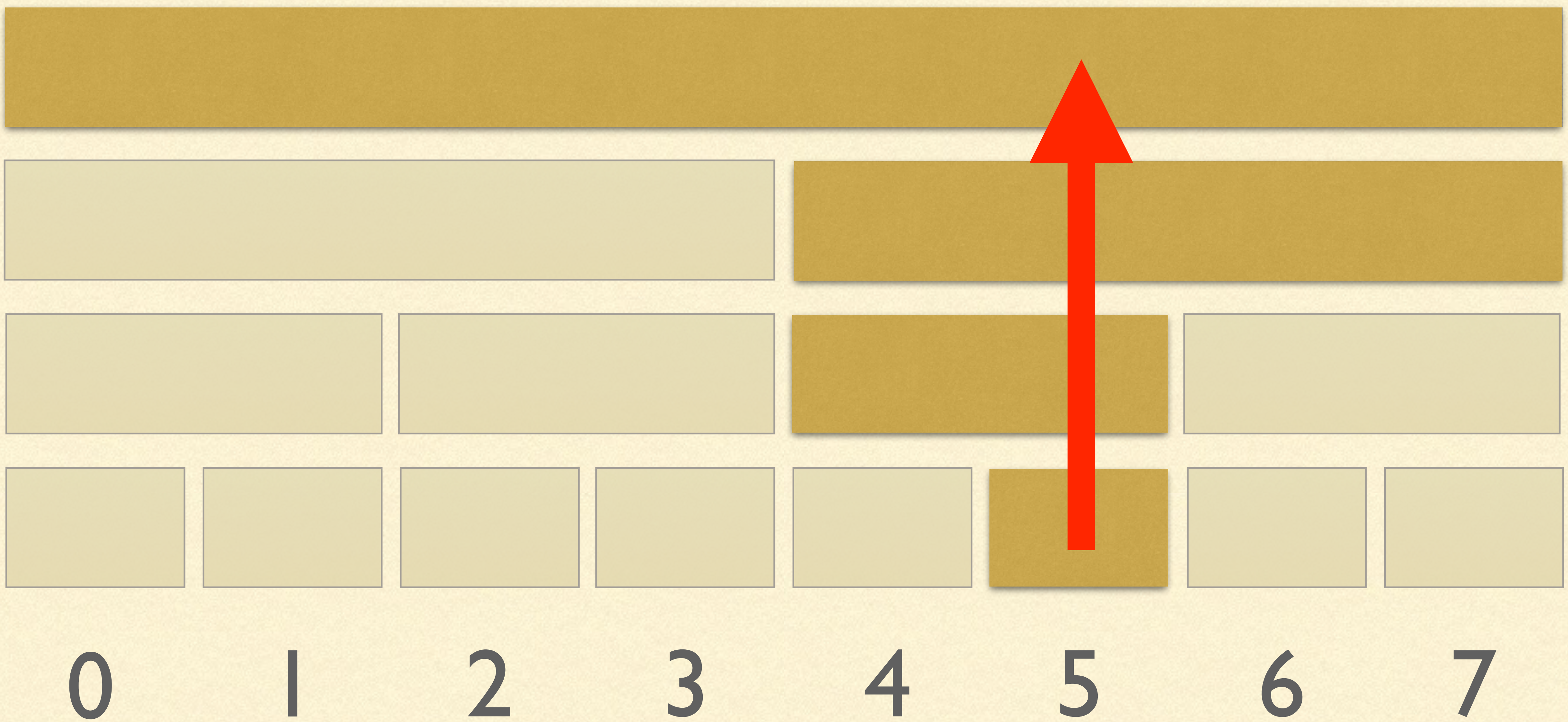

1
10
11
100
101
110
111
1000
1001
1010
1011
1100
1101
1110
1111
0
1
2
3
4
5
6
7

左の子を見るときは、２倍する

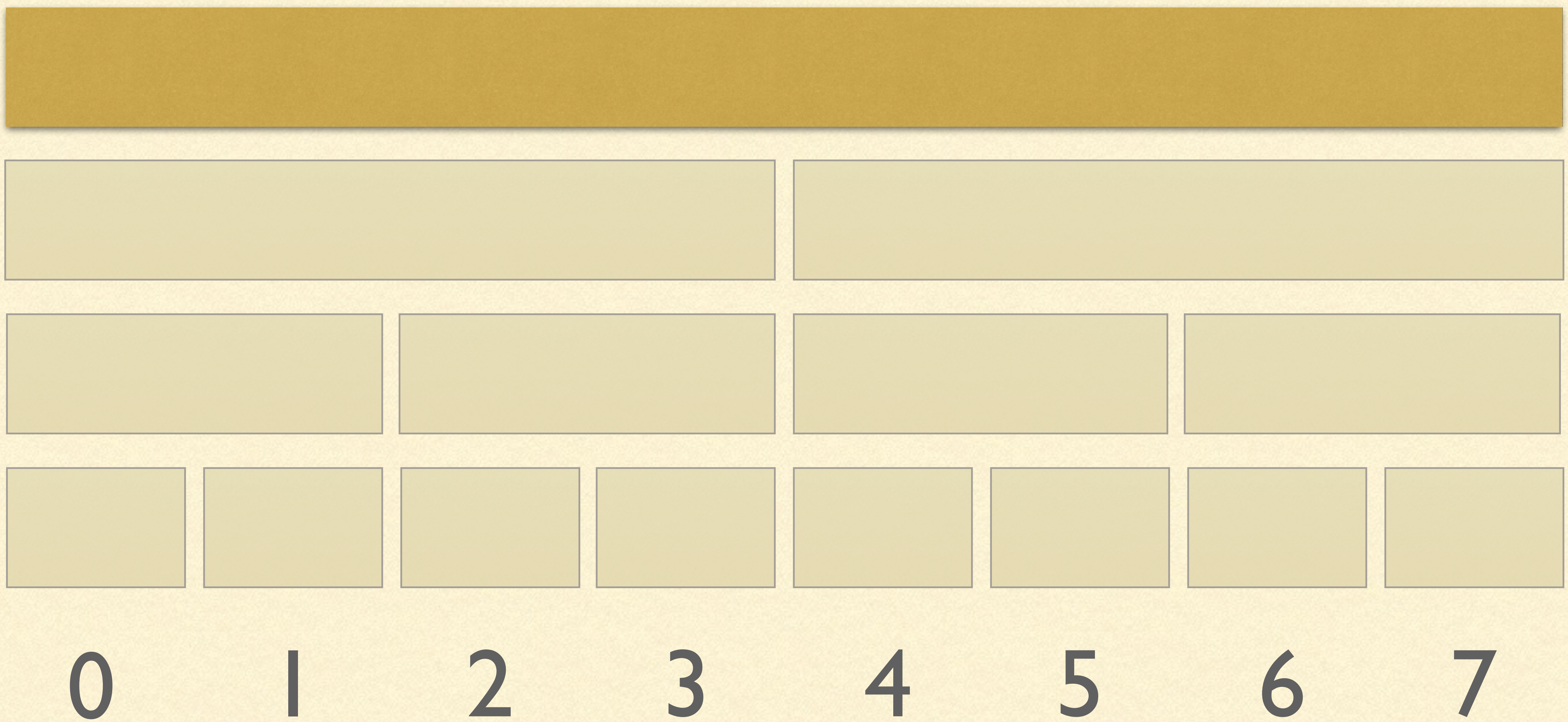
右の子を見るときは、２倍して１足す

親を見るときは、２で割る

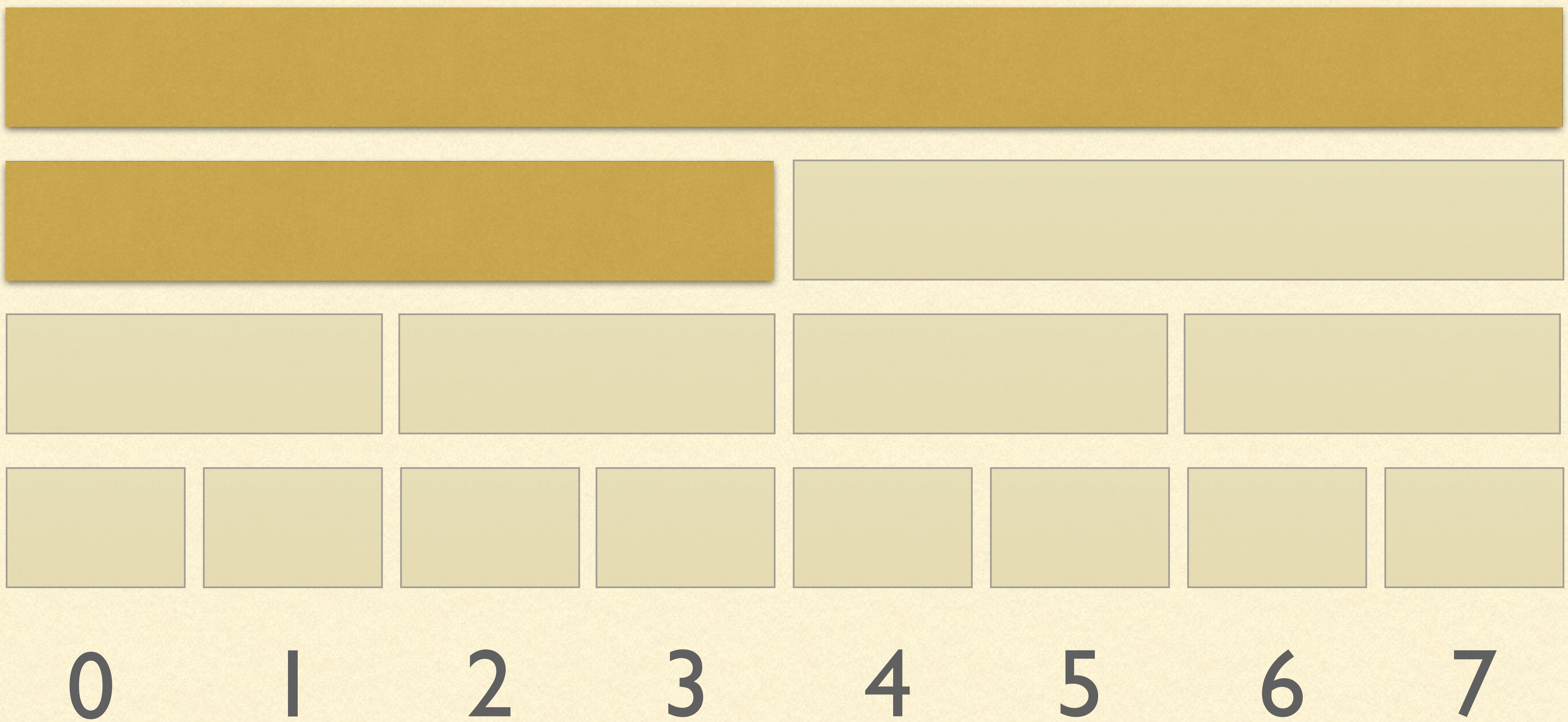
“差の要素”に5を追加／削除するときの更新対象



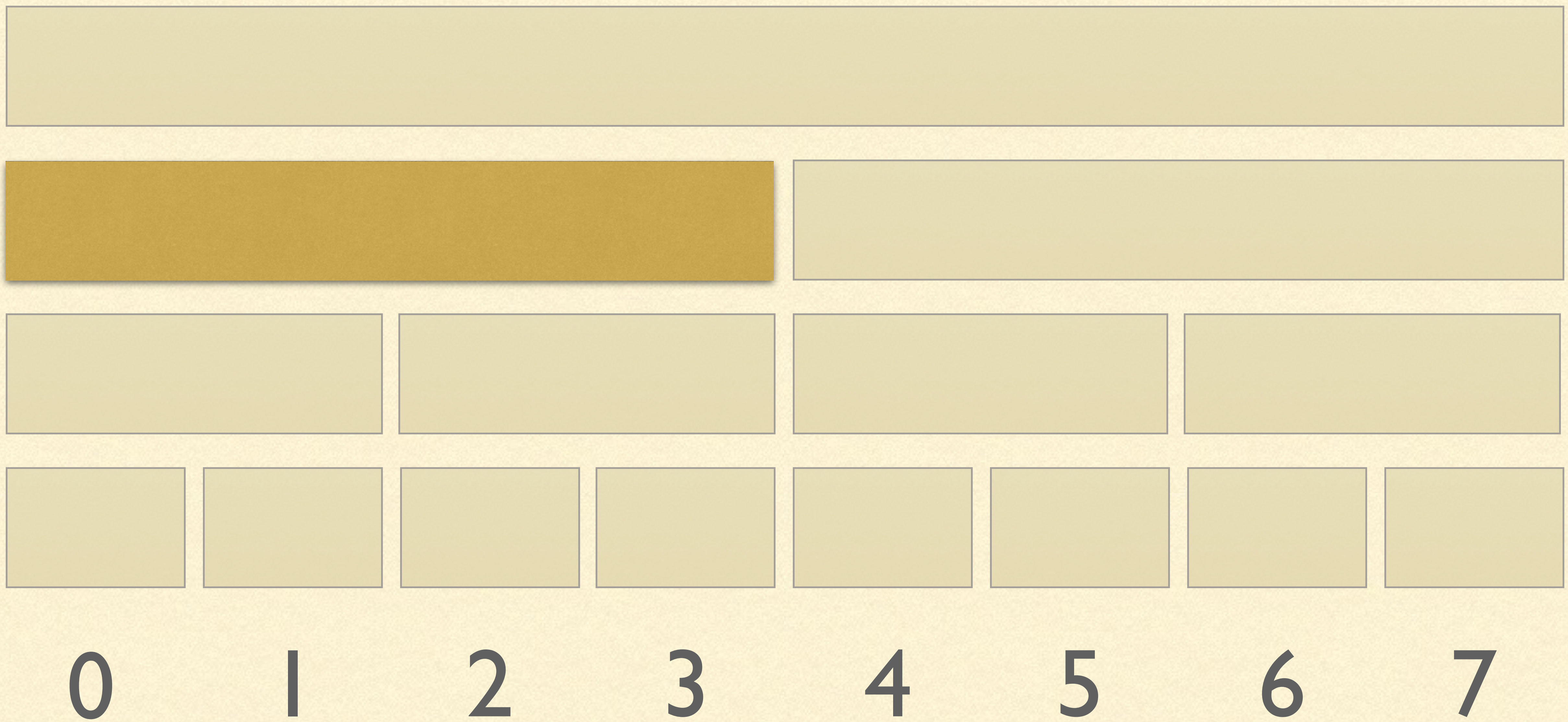
「小さい方からX個の合計」を求める



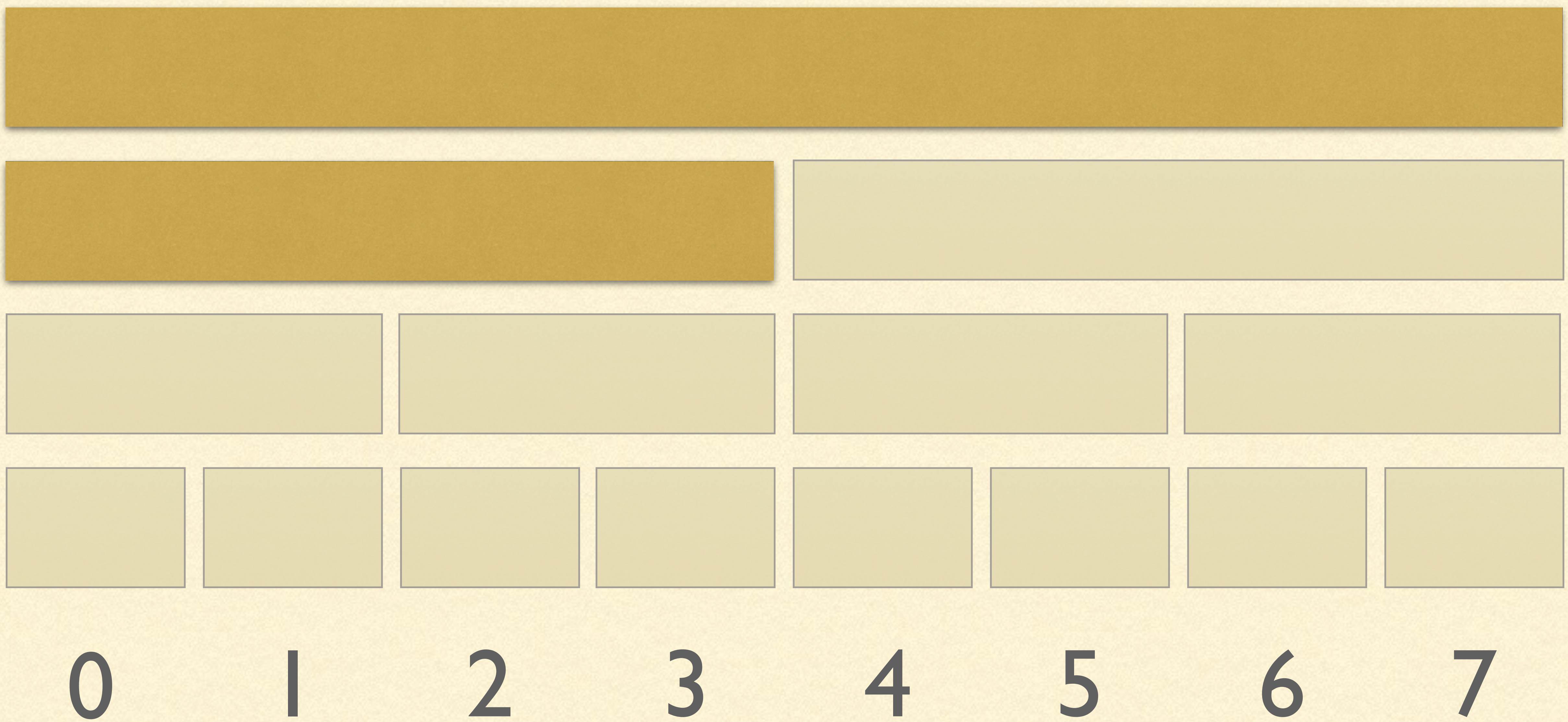
左の子（0〜3）がX個以上持っていれば、



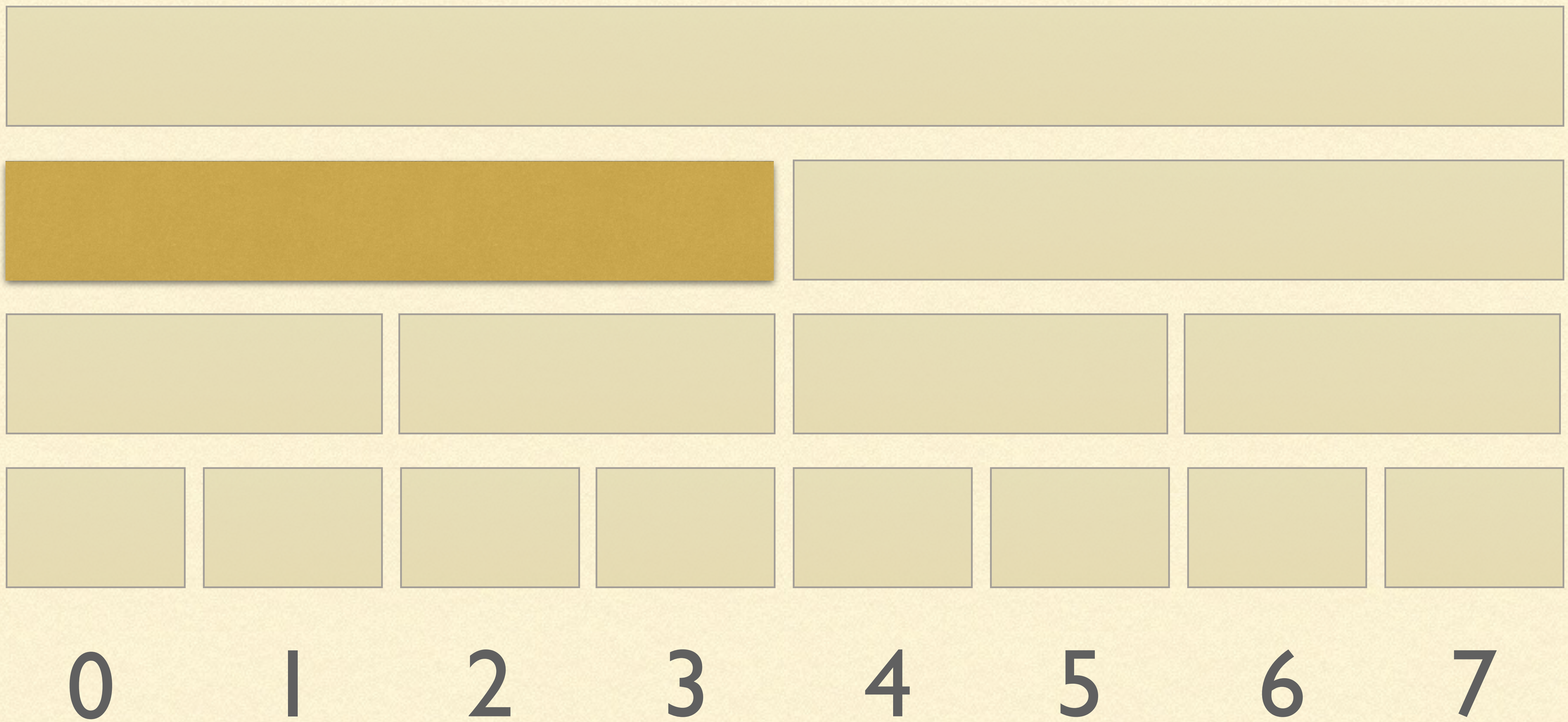
左の子（0〜3）がX個以上持っていれば、
再帰的に左の子より下を引き続き見る



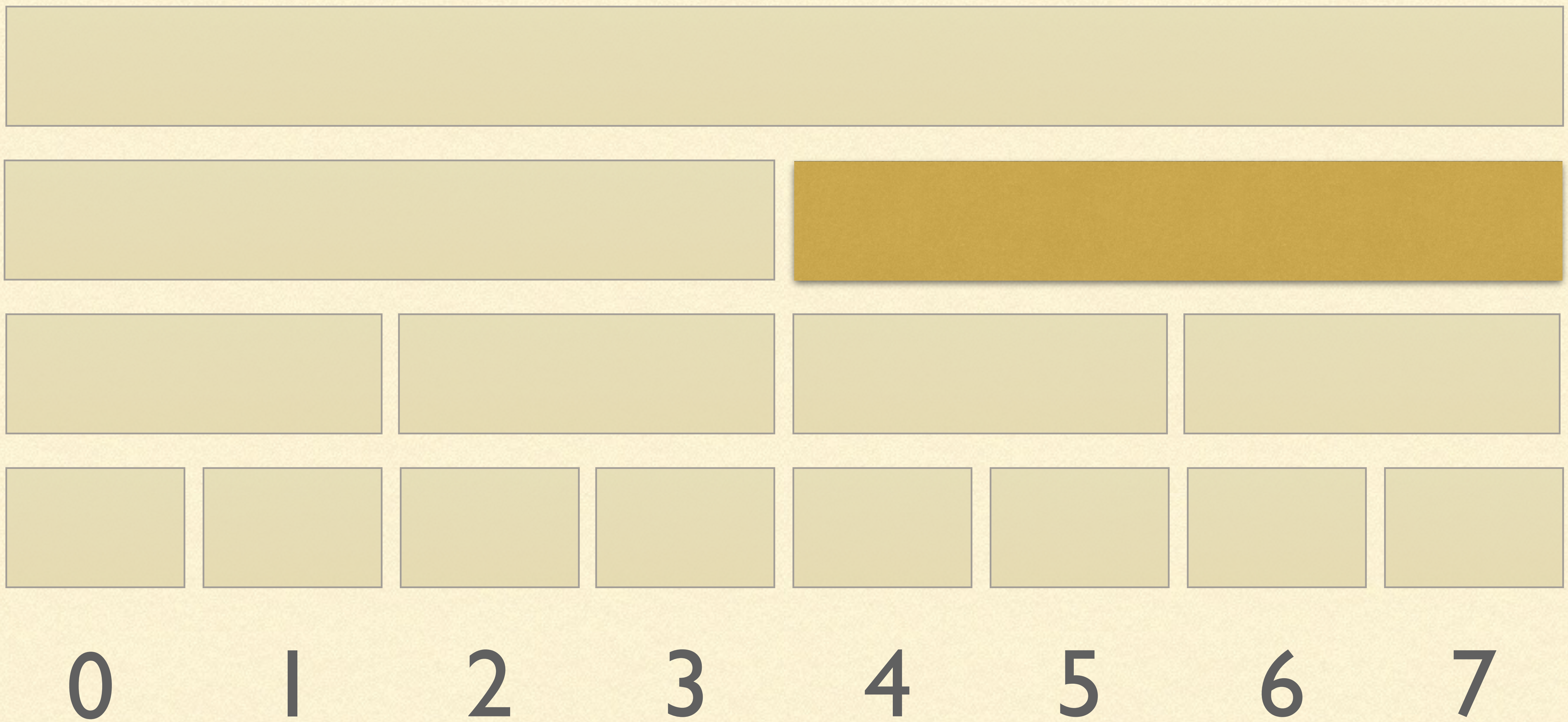
左の子（0〜3）がX個未満（Y個）であれば、



左の子（0〜3）がX個未満（Y個）であれば、
左の子（0〜3）の合計値は必ず使用し、



左の子（0〜3）がX個未満（Y個）であれば、
左の子（0〜3）の合計値は必ず使用し、
右の子（4〜7）の小さい方からX-Y個を取得



問題D　満点解法（別解）

- “差の要素”の小さい方から**n-k-i**個の和が必要になるので、
-
-

問題D　満点解法（別解）

- “差の要素”の小さい方からn-k-i個の和が必要になるので、
- n-k-i個以下のmultisetと、
  n-k-i個超のmultisetの２つを用意する
- “差の要素”の追加・削除に応じて、
  ２つのmultiset間でやりとりする。

ありがとうございました！