2013-2014年JOI本選問題第3問
バームクーヘン

笠浦一海

問題

- バームクーヘンを三つに分ける
- 切れるところは決まってる
- 三つのうち大きさが最小のピースの大きさを最大化したい

愚直な方法

- 全探索
- 切る部分を決める→約N^3/6通り
- それぞれのピースの大きさを求める

→愚直に合計するとO(N)
- 合わせてO(N^4)
- Subtask1 N<=100なら解ける→5点

注意点

- 各部分のおおきさA_iの条件
- 1≦A_i≦1000000000
- intの範囲ぎりぎり
- **オーバーフローに注意**
- long longを使うべき

注意点

- 円環上の数列を扱う。
- Aの後ろにAのコピーをつなげて
- $A_{N+1}=A_1,\ldots,A_{N+i}=A_i,\ldots,A_{2N}=A_N$
- とすると実装上便利

注意点

- 切れ込み1を含むようなピースの大きさ
- $A_i \ldots A_N, A_1 \ldots A_j$の合計を求めるとき

```
long long Sum=0;

for(int k=i;k<=N;k++)
   Sum+=A[k];
for(int k=1;k<=j;k++)
   Sum+=A[k];
```

注意点

- 切れ込み1を含むようなピースの大きさ
- $A_i \ldots A_N, A_1 \ldots A_j$の合計を求めるとき

```
long long Sum=0;

for(int k=i;k<=j+N;k++)
    Sum+=A[k];
```

もう少しましな方法

- 前の方法ではいちいち合計を求めている。
- 同じ区間について何度も合計を求めていて無駄。
- 区間の数O(N^2)個についてあらかじめ合計を求めて記憶しておく。
- 前処理O(N^3)で、それぞれの切り方についてしらべる段階でもO(N^3)
- 全体計算量O(N^3)
- Subtask2 N<=400までならOK→ 20点

もう少しましな方法２

- 累積和
- $A_1 \ldots A_i$の合計をB_iとおく。
- 漸化式
- $B_0=0$
- $B_{i+1}=B_i+A_{i+1}$
- でO(N)で計算可能。

もう少しましな方法2

- 累積和
  - $A_i \ldots A_j$の合計$=B_j-B_{i-1}$
  - $A_i \ldots A_N, A_1 \ldots A_j$の合計$=B_N-B_{i-1}+B_j=B_{j+N}-B_{i-1}$
  - と各ピースの大きさを**O(1)**で求められる。
  - これでも計算量**O(N^3)**になる。
  - 以下のスライドでも区間の合計は定数時間で求められるとする

解法

- これ以上の点数を取るのに必要なテクニック
- 二分探索
- 尺取り法

解で二分探索

- ①「3つのピースの大きさの最小値の最大値を求めよ」
- という問題を次の判定問題に還元する
- ②「3つのピースとも大きさが**X**以上になる切り方は存在するか？」
- ②が解けたとする

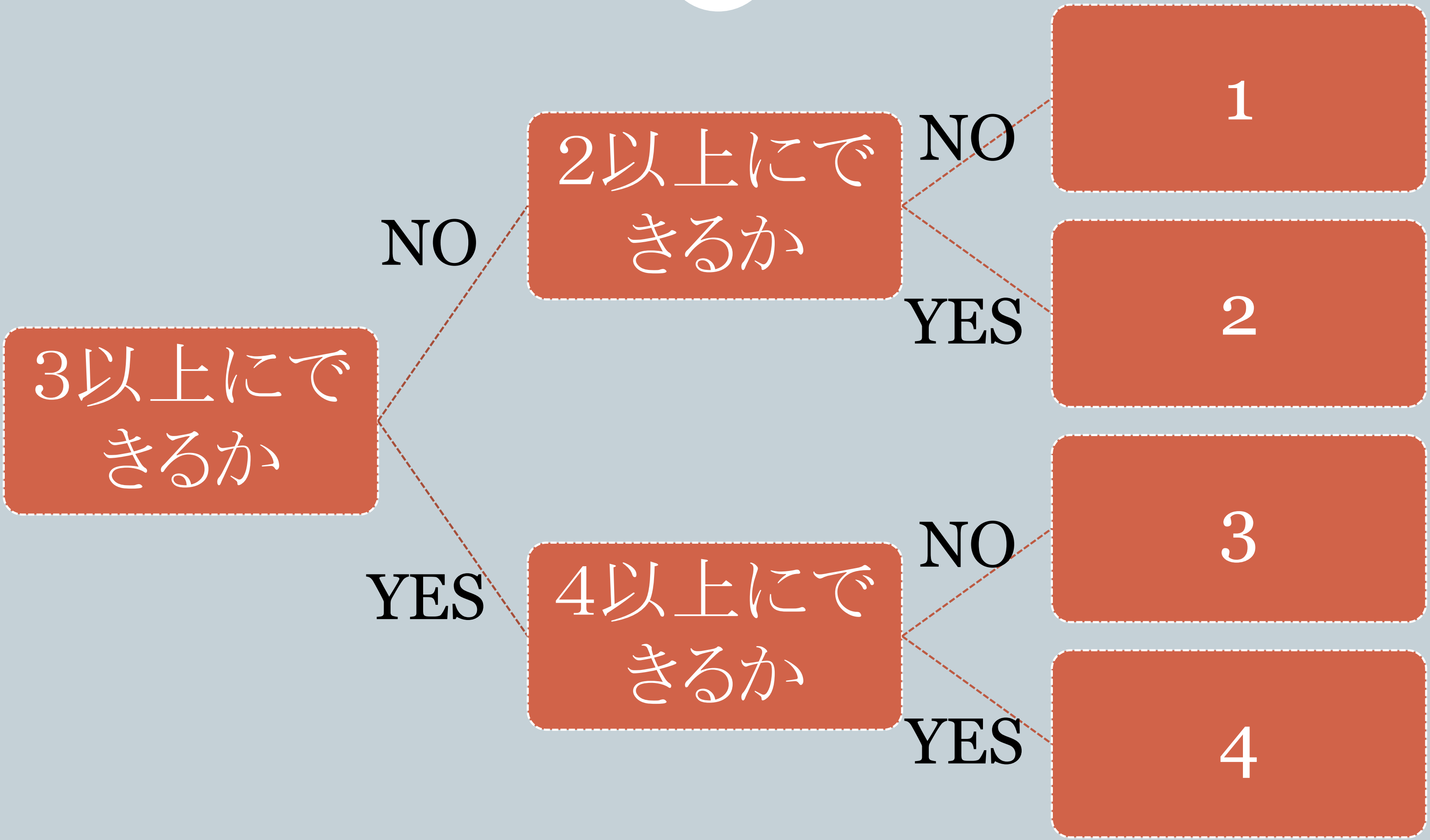
解で二分探索

- 求めたい最大値の上限を適当に定めてMAXとする。
- この問題ではMax=1000000000*Nとすればよい。
- 最大値は[1,Max+1)の範囲に存在。
- m=(Max+2)/2とする。
- X=mについて②を解く。
- Yes→求める値は[m,Max+1)の範囲に存在。
- No→求める値は[1,m)の範囲に存在。

解で二分探索

- これを再帰的に行う。
- 求めたい値が[l,r)の範囲に存在することが分かっているとする。
- m=(l+r)/2とする。
- X=mについて②を解く。
- Yes→求める値は[m,r)の範囲に存在。
- No→求める値は[l,m)の範囲に存在。
- r-l=1となるまで繰り返す。

例(Max=4)



解で二分探索

- すなわち②が解ければ①は
- (②を解くのにかかる時間)×log Max
- で解くことができる。

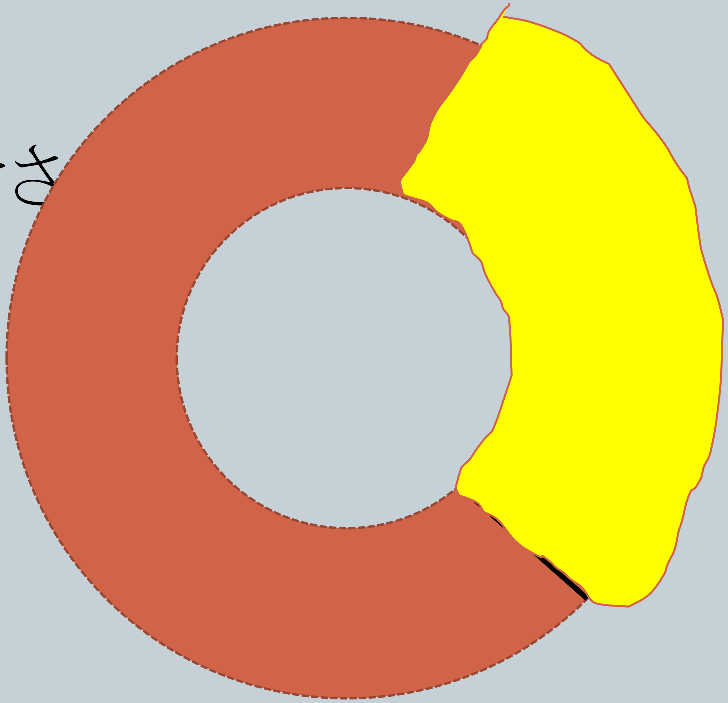
②の解き方

- 「どのピースも大きさが**X**以上になるように切ることができるか？」
- まず切る位置を一つ決める。
- 次に切る位置を決めたい。
- 両者の間のピース(黄)の大きさ

X以上ならよい

- 逆に**X**以上ならそれより

大きくしても無意味



②の解き方

- 「どのピースも大きさが**X**以上になるように切ることができるか？」
- 間の部分の大きさが最初に

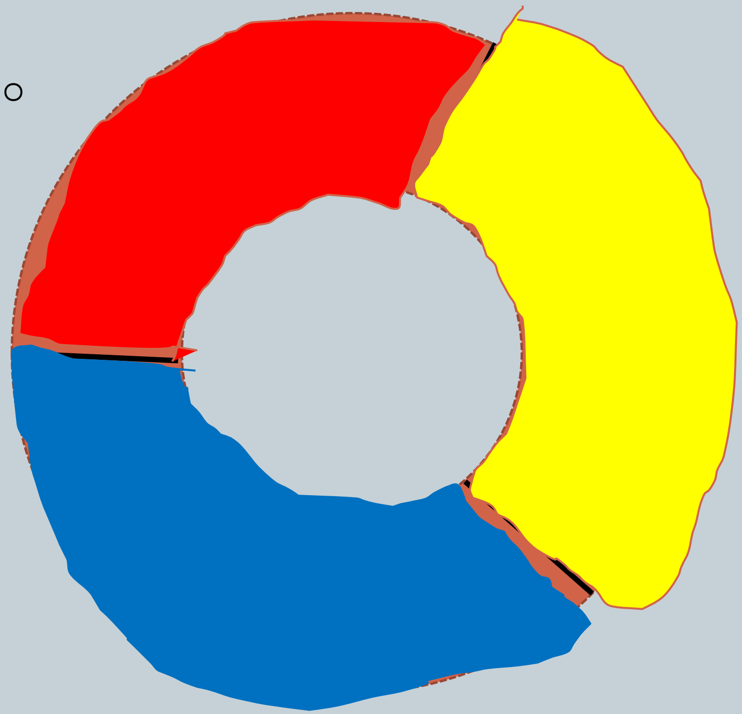
Xを超える位置できるのが最善。

順番に見ていけばそのような

位置は**O(N)**で見つかる。

- もう**1**か所も同様に切る。
- 残りの部分**(**赤色**)**の大きさが

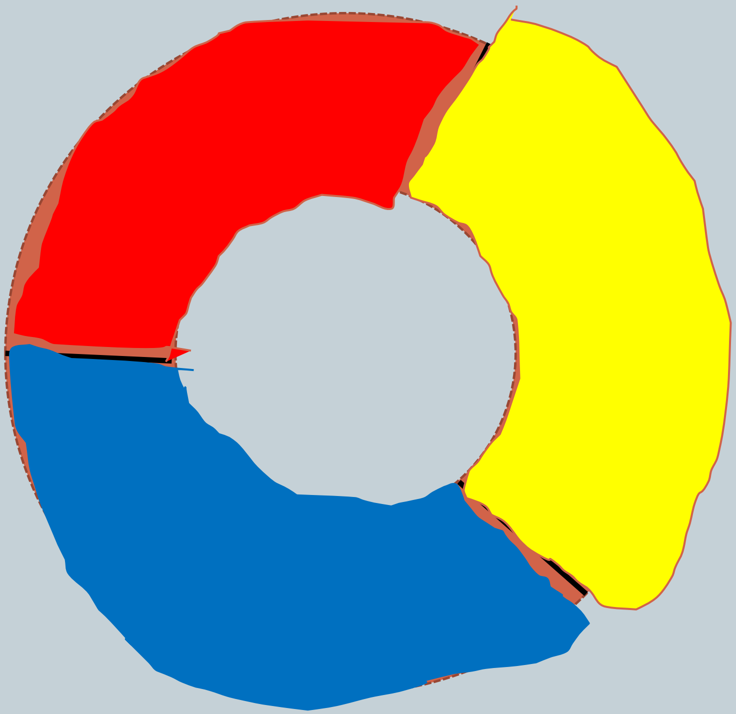
X以上なら成功。



②の解き方

- 「どのピースも大きさがX以上になるように切ることができるか？」
- 切る位置N通りについて調べる。
- どれか一つで成功ならYes
- 成功がないならNo
- ②がO(N^2)で解ける。
- ①はO(N^2 log Max)
- Subtast 3 N<=8000

は少し厳しい。



②の早い解き方(二分探索)

- 「どのピースも大きさがX以上になるように切ることができるか？」
- 「最初にXを超える位置」を二分探索で求める。
- 求めたいのは

$$B_l - B_{i-1} >= X$$

となるlの最小値
- 自分で二分探索を書かなくてもlower_boundが使える。

②の早い解き方(二分探索)

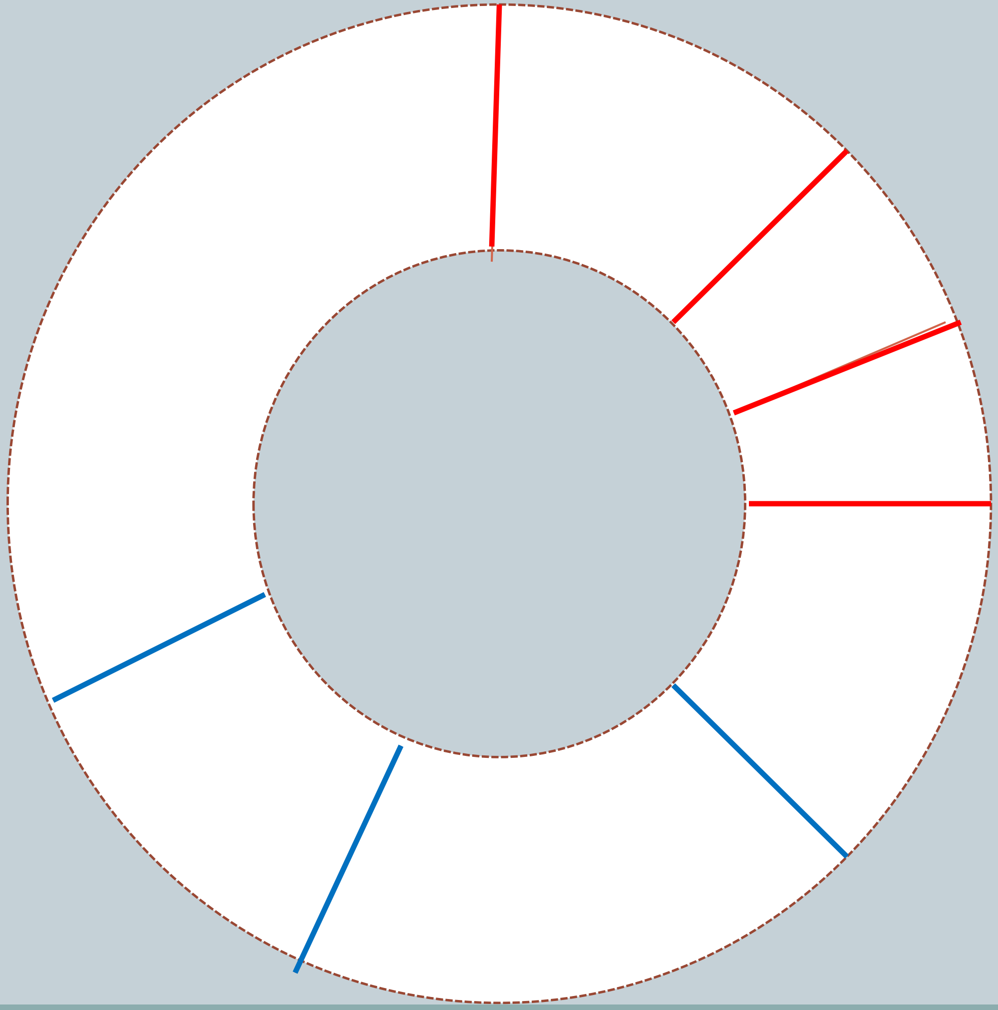
- 「どのピースも大きさがX以上になるように切ることができるか？」
- こうするとO(N logN)で②が解ける
- ①の計算量はO(N log N log Max)となる。
- N<=100000なら大丈夫→満点！

②の早い解き方(尺取り法)

- 「どのピースも大きさがX以上になるように切ることができるか？」
- iについて$B_l - B_{i-1} >= X$なるlの最小値をl(i)と書く。
- $l(i+1) >= l(i)$
- なぜならば$B_{l(i)-1} - B_i <= B_{l(i)-1} - B_{i-1} < X$より$l(i)-1 < l(i+1)$
- l(i+1)を調べるときはl(i)から見ていけばよい。
- これで実はO(N)になる。

②の早い解き方(尺取り法)

$X=135^{\circ}$



②の早い解き方(尺取り法)

- 尺取り虫っぽい動きをしている→これを尺取り法という
- l(i)が単調に増加するので値が増える回数は高々2N
- O(N)ですべてのiに対するl(i)計算ができる。
- l(i)を計算しておけば②がO(N)で解ける。
- ①がO(N log Max)で解ける→満点

別解

- 「ある区間について、それが最小のピースとなりうるか？」という問題を考える。
- その区間(黄色)のサイズをXとする。
- もうひとつのピース(青色)

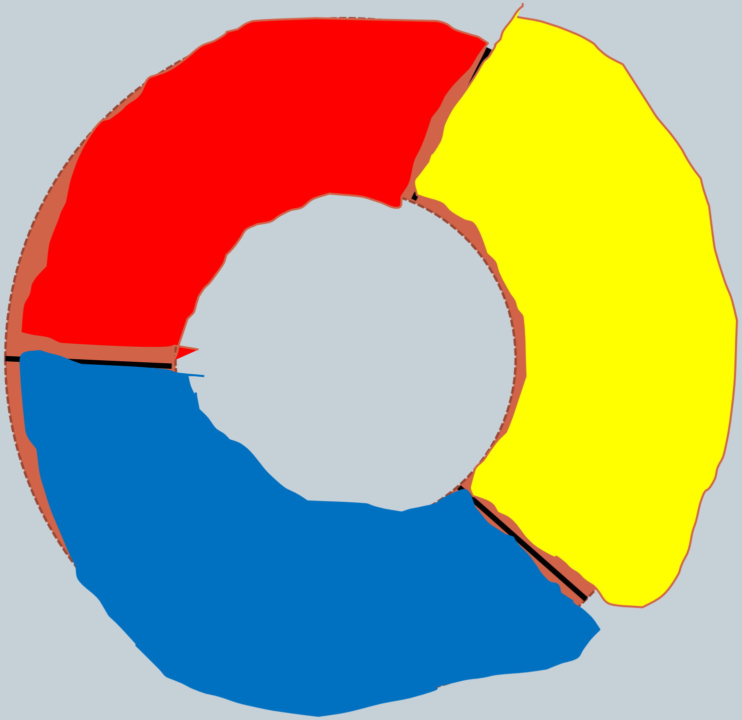
の部分の大きさがX以上になる

最初のところで切る

- lower_boundが使える。
- 残った赤色のピースの

大きさがX以上なら可能。

- そうでないなら不可能。



別解

- すべての区間について調べて、最小のピースとなりえるものの最大値を取る。
- 計算量はO(N^2 log N)
- 定数が軽いので適当に枝刈りすれば

Subtask 3: N<=8000 もぎりぎり通りそう。

- 50点?

別解

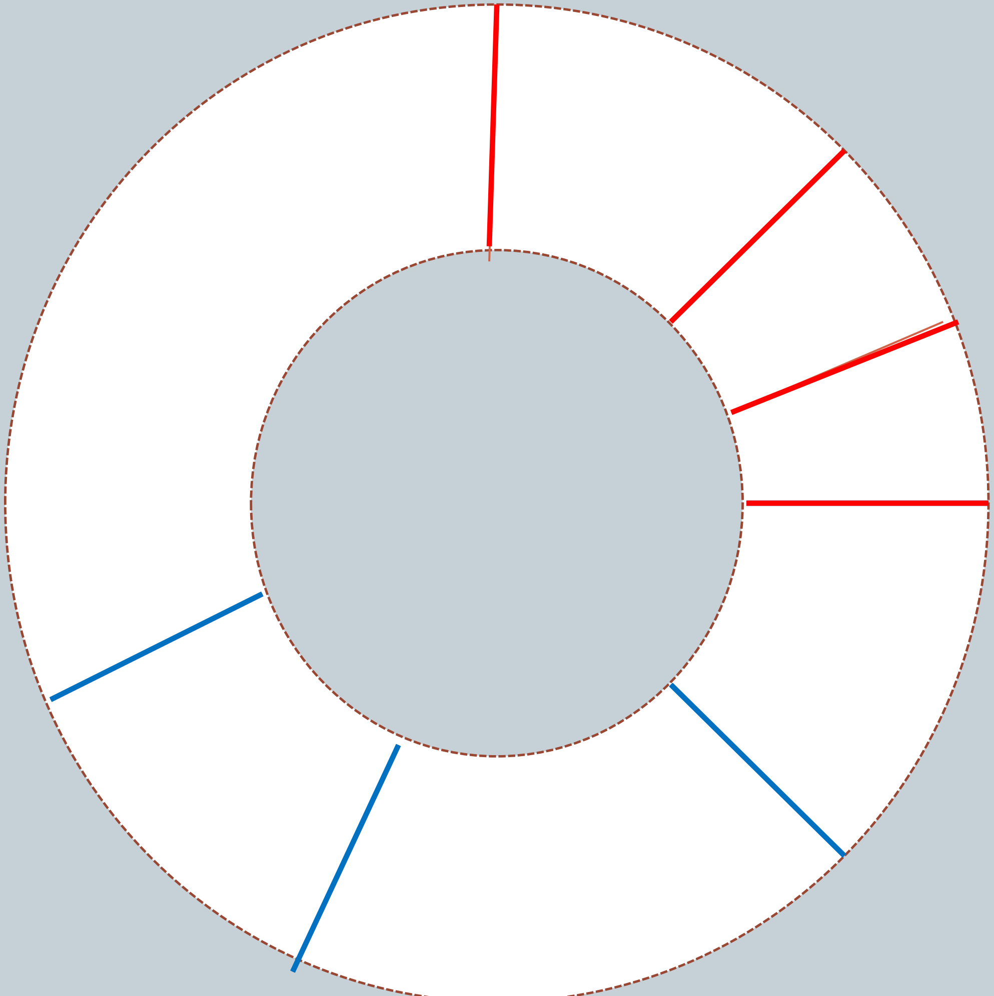
- 各iについてA_i～A_mの区間が最小のピースになりうるようなlの最大値をm(i)とする。
- 求めたいのは$B_{m(i)}$-B_{i-1}の最大値。
- m(i+1)>=m(i)となる。

→先ほどと同様に尺取り法が使える！

- 条件を満たすかの判定にO(log N)かかるので、全体の計算量はO(N log N)になる。

→満点！

別解



二分探索を使わない方法

- 尺取り法だけで解きたい。
- さっきの方法だと3つ目の切る位置が単調に変化しないので尺取りできない。
- そこで3つ目の切る位置を、残りの部分をできるだけ2等分に近く切る点とする。すなわち、小さいほうのピースの大きさを最大化するところできる。
- この方法でも「ある区間について、それが最小のピースとなりうるか？」の判定ができる。
- こうすると**O(N)**で解くことができる。

二分探索を使わない方法