

ARC 120 A - Max Add

hiragn

2024 年 12 月 18 日

1. 問題の概要

数列 $a = (a_1, a_2, \dots, a_k)$ に以下の操作を行った後の a の要素の総和を $f(a)$ とする。
 (操作) $i = 1, 2, \dots, k$ の順に現在の a の要素の最大値を a_i に足す
 長さ N の数列 $A = (A_1, A_2, \dots, A_N)$ が与えられる。
 $a = (A_1, A_2, \dots, A_k)$ ($1 \leq k \leq N$) としたときの $f(a)$ の値 N 個を求めよ。

https://atcoder.jp/contests/arc120/tasks/arc120_a

2. 解法

実験。 $a = (1, 3, 5, \dots, 2n-1)$ とする。

$(1, 3, 5, \dots, 2n-1)$
 $\rightarrow (2n, 3, 5, \dots, 2n-1)$
 $\rightarrow (2n, 2n+3, 5, \dots, 2n-1)$
 $\rightarrow (2n, 2n+3, 2n+8, \dots, 2n-1)$
 $\rightarrow \dots$

i 回操作した後の最大値を m_i とおく。特に $m_0 = \max(a)$ とする。

数列の各項は $a_i \rightarrow a_i + m_{i-1}$ のように変化し、最大値は毎回更新される。

$$m_i = \max\{m_{i-1}, a_i + m_{i-1}\} = a_i + m_{i-1}$$

変換操作後の数列を $b = (b_1, b_2, \dots, b_n)$ とする。

$$\begin{aligned}
 b_i &= a_i + m_{i-1} = a_i + (a_{i-1} + m_{i-2}) \\
 &= \dots = \sum_{k=1}^i a_k + m_0
 \end{aligned}$$

$f(a)$ はこれの和。 a の累積和の数列を c とする。

$$f(a) = \sum_{i=1}^n b_i = \sum_{i=1}^n c_i + n \cdot m_0$$

以上は数列 a を最後まで見たときの話。 k 個目まで見たときは m_0 を「 $a_1 \sim a_k$ の最大値」に変えればいい。これは `FoldList` で作れる。

```
m = FoldList[Max, First@lst, Rest@lst];
```

1 項目は 1 倍, 2 項目は 2 倍, ... は `Times` を `Thread` にわたすことで実現できる。

```
Thread[Times[m, Range@Length@m]];
```

$\sum_{i=1}^n c_i$ は累積和を 2 回とると高速に求められる。これと上の `Thread` の和が答え。

```
1 In[] := Clear["Global`*"];
2 solve[lst_] := Module[{m},
3   m = FoldList[Max, First@lst, Rest@lst];
4   Thread[Times[m, Range@Length@m]] + Accumulate@Accumulate@lst];
5
6 case = {1, 2, 3};
7 res = {2, 8, 19};
8 solve@case == res
9
10 Out[] = True
```
