

[BZOJ3675][APIO2014]序列分割sequence

2014-07-23 22:16:08 | 分类: [BZOJ](#) | 标签: [dp](#) [斜率优化](#) [二分](#)

[订阅](#) | [字号](#) | [举报](#)

[我的照片书](#) | [下载LOFTER](#)

3675: [APIO2014]序列分割

Time Limit: 40 Sec Memory Limit: 128 MB

Submit: 595 Solved: 222

[\[Submit\]](#)[\[Status\]](#)[\[Discuss\]](#)

Description

小H最近迷上了一个分割序列的游戏。在这个游戏里，小H需要将一个长度为N的非负整数序列分割成 $k+1$ 个非空的子序列。为了得到 $k+1$ 个子序列，小H将重复进行七次以下的步骤：

1. 小H首先选择一个长度超过1的序列（一开始小H只有一个长度为n的序列——也就是一开始得到的整个序列）；
2. 选择一个位置，并通过这个位置将这个序列分割成连续的两个非空的新序列。

每次进行上述步骤之后，小H将会得到一定的分数。这个分数为两个新序列中元素和的乘积。小H希望选择一种最佳的分割方案，使得k轮（次）之后，小H的总得分最大。

Input

输入文件的第一行包含两个整数n和k($k+1 \leq n$)。

第二行包含n个非负整数 a_1, a_2, \dots, a_n ($0 \leq a_i \leq 10^4$)，表示一开始小H得到的序列。

Output

一行包含一个整数，为小H可以得到的最大得分。

Sample Input

```
7 3
4 1 3 4 0 2 3
```

Sample Output

```
108
```

HINT

【样例说明】

在样例中，小H可以通过如下3轮操作得到108分：

1. 开始小H有一个序列(4, 1, 3, 4, 0, 2, 3)。小H选择在第1个数之后的位置将序列分成两部分，并得到 $4 \times (1+3+4+0+2+3) = 52$ 分。
2. 这一轮开始时小H有两个序列：(4), (1, 3, 4, 0, 2, 3)。小H选择在第3个数 字之后的位置将第二个序列分成两部分，并得到 $(1+3) \times (4+0+2+3) = 36$ 分。
3. 这一轮开始时小H有三个序列：(4), (1, 3), (4, 0, 2, 3)。小H选择在第5个数字之后的位置将第三个序列分成两部分，并得到 $(4+0) \times (2+3) = 20$ 分。

经过上述三轮操作，小H将会得到四个子序列：(4), (1, 3), (4, 0), (2, 3)并总共得到 $52+36+20=108$ 分。

【数据规模与评分】

数据满足 $2 \leq n \leq 100000, 1 \leq k \leq \min(n-1, 200)$ 。

Solution

看上去像斜率优化就写了。

记 $f[i][s]$ 表示做到第 i 个数，前面割了 s 刀的最大获利。 $sum[i]$ 表示 $\sum_{a[1..i]}$ 。

显然 $f[i][s] = \max(f[j][s-1] + sum[j] * (sum[i] - sum[j])) (j < i)$

设 $k < j$ ，并且 j 比 k 优，有：

$$f[j][s-1] + sum[j] * (sum[i] - sum[j]) \geq f[k][s-1] + sum[k] * (sum[i] - sum[k])$$

$$\text{化简得 } \frac{sum^2[k] - f[k][s-1] - (sum^2[j] - f[j][s-1])}{sum[k] - sum[j]} \geq sum[i]$$

记 $y[i][s] = sum^2[i] - f[i][s]$ ，则 i 号位置对应点 $(sum[i], y[i][s])$

由于斜率越大越好，因此在队列中的斜率单调递增。用单调队列维护即可。

oj丧病卡内存所以要把 f 和 y 滚动掉。

upd-20150328

这题等价于将序列分割成 $k+1$ 段，使得每段的平方和最小。

设 $f[i]$ 表示做到第 i 个数，前面割了？刀的最小代价，

则有 $f[i] = \min(f[j] + (sum[i] - sum[j])^2) (j < i)$

显然如果不加任何东西直接这样算的话， $f[n] = a_1^2 + a_2^2 + a_3^2 + \dots + a_n^2$ ，因为二次函数是下凹函数。

怎么办呢？机智的AKF同学参考了CLJ当年出过的tree的方法，把 $f[i]$ 改成这样：

$$f[i] = \min(f[j] + (sum[i] - sum[j])^2) + C$$

显然当 $C = +\infty$ 时，答案必定是只分一段，当 $C = 0$ 时，显然答案是每个数割一刀，分成 n 段。

并且当 C 取遍 $[0, +\infty)$ 时，刀函数是连续的。因此可以二分。剩下的还是斜率优化花 $O(n)$ 时间判定。

所以其实这题 k 可以与 n 同阶