

序列查询新解

时间限制：1.0 秒

空间限制：512 MiB



题目背景

上一题“序列查询”中说道： $A = [A_0, A_1, A_2, \dots, A_n]$ 是一个由 $n + 1$ 个 $[0, N)$ 范围内整数组成的序列，满足 $0 = A_0 < A_1 < A_2 < \dots < A_n < N$ 。基于序列 A ，对于 $[0, N)$ 范围内任意的整数 x ，查询 $f(x)$ 定义为：序列 A 中**小于等于** x 的整数里**最大的**的数的**下标**。

对于给定的序列 A 和整数 x ，查询 $f(x)$ 是一个很经典的问题，可以使用二分搜索在 $O(\log n)$ 的时间复杂度内轻松解决。但在 IT 部门讨论如何实现这一功能时，小 P 同学提出了些新的想法。

题目描述

小 P 同学认为，如果事先知道了序列 A 中整数的分布情况，就能直接估计出其中小于等于 x 的最大整数的大致位置。接着从这一估计位置开始线性查找，锁定 $f(x)$ 。如果估计得足够准确，线性查找的时间开销可能比二分查找算法更小。

比如说，如果 A_1, A_2, \dots, A_n 均匀分布在 $(0, N)$ 的区间，那么就可以估算出：

$$f(x) \approx \frac{(n+1) \cdot x}{N}$$

为了方便计算，小 P 首先定义了比例系数 $r = \left\lfloor \frac{N}{n+1} \right\rfloor$ ，其中 $\lfloor \cdot \rfloor$ 表示下取整，即 r 等于 N 除以 $n + 1$ 的商。进一步地，小 P 用 $g(x) = \left\lfloor \frac{x}{r} \right\rfloor$ 表示自己估算出的 $f(x)$ 的大小，这里同样使用了下取整来保证 $g(x)$ 是一个整数。

显然，对于任意的询问 $x \in [0, N)$ ， $g(x)$ 和 $f(x)$ 越接近则说明小 P 的估计越准确，后续进行线性查找的时间开销也越小。因此，小 P 用两者差的绝对值 $|g(x) - f(x)|$ 来表示处理询问 x 时的误差。

为了整体评估小 P 同学提出的方法在序列 A 上的表现，试计算：

$$\text{error}(A) = \sum_{i=0}^{N-1} |g(i) - f(i)| = |g(0) - f(0)| + \dots + |g(N-1) - f(N-1)|$$

输入格式

从标准输入读入数据。

输入的第一行包含空格分隔的两个正整数 n 和 N 。

输入的第二行包含 n 个用空格分隔的整数 A_1, A_2, \dots, A_n 。

注意 A_0 固定为 0，因此输入数据中不包括 A_0 。

输出格式

输出到标准输出。
仅输出一个整数，表示 $error(A)$ 的值。

样例1输入

3 10
2 5 8

样例1输出

5

样例1解释

$$A = [0, 2, 5, 8]$$
$$r = \left\lfloor \frac{N}{n+1} \right\rfloor = \left\lfloor \frac{10}{3+1} \right\rfloor = 2$$

i	0	1	2	3	4	5	6	7	8	9
$f(i)$	0	0	1	1	1	2	2	2	3	3
$g(i)$	0	0	1	1	2	2	3	3	4	4
$\ g(i) - f(i)\ $	0	0	0	0	1	0	1	1	1	1

样例2输入

9 10
1 2 3 4 5 6 7 8 9

样例2输出

0

样例3输入

2 10
1 3

样例3输出

6

样例3解释

$$A = [0, 1, 3]$$
$$r = \left\lfloor \frac{N}{n+1} \right\rfloor = \left\lfloor \frac{10}{2+1} \right\rfloor = 3$$

i	0	1	2	3	4	5	6	7	8	9
$f(i)$	0	1	1	2	2	2	2	2	2	2
$g(i)$	0	0	0	1	1	1	2	2	2	3
$\ g(i) - f(i)\ $	0	1	1	1	1	1	0	0	0	1

子任务

70% 的测试数据满足 $1 \leq n \leq 200$ 且 $n < N \leq 1000$;

全部的测试数据满足 $1 \leq n \leq 10^5$ 且 $n < N \leq 10^9$ 。

提示

需要注意，输入数据 $[A_1 \cdots A_n]$ 并不一定均匀分布在 $(0, N)$ 区间，因此总误差 $error(A)$ 可能很大。

语言及编译选项信息

#	名称	编译器	额外参数	代码长度限制 (B)
0	g++ with std11	g++	-O2 -std=c++11 - DONLINE_JUDGE	65536
1	g++	g++	-O2 - DONLINE_JUDGE	65536

#	名称	编译器	额外参数	代码长度限制 (B)
2	gcc with std11	gcc	-O2 -std=c11 - DONLINE_JUDGE	65536
3	gcc	gcc	-O2 - DONLINE_JUDGE	65536
4	java	javac		65536
5	python	python		65536
6	python3	python3		65536

递交历史

#	状态	时间
9840	<div>Running</div> (/#!/contest/13/detail/9840)	02:54:41 PM <div>有效递交</div>
9839	<div>Running on case 1</div> (/#!/contest/13/detail/9839)	02:54:39 PM
9834	<div>Running on case 7</div> (/#!/contest/13/detail/9834)	02:54:28 PM
9823	<div>Running on case 15</div> (/#!/contest/13/detail/9823)	02:54:17 PM
9822	<div>Running on case 17</div> (/#!/contest/13/detail/9822)	02:54:15 PM

#	状态	时间
9820	Running on case 17 (/#!/contest/13/detail/9820)	02:54:14 PM
9817	Wrong Answer (/#!/contest/13/detail/9817)	02:54:11 PM
9816	Wrong Answer (/#!/contest/13/detail/9816)	02:54:11 PM
9815	Wrong Answer (/#!/contest/13/detail/9815)	02:54:10 PM
9810	Running on case 14 (/#!/contest/13/detail/9810)	02:54:08 PM
		<div>1234</div>

递交答案 (剩余次数: 32)

语言和编译选项

g++ with std11

▼

1

递交评测

文件请拖入编辑器中，或

上传文件