

NOIP 模拟

题目列表 Problem List

题目名称	按位排序	鲍勃的数学题	最大子段和	鲍勃打比赛
输入文件名	bitsort.in	math.in	subarray.in	contest.in
输出文件名	bitsort.out	math.out	subarray.out	contest.out
每个测试点时限	2s	3s	3s	4s
测试点数目	10	10	10	10
每个测试点分值	10	10	10	10
内存限制	256MB	256MB	512MB	256MB
题目类型	传统	传统	传统	传统

请勿在比赛开始前翻阅试题!

Do not open before the contest has started.

Problem A. 按位排序

Input file: bitsort.in
Output file: bitsort.out
Time limit: 2 second
Memory limit: 256 megabytes

鲍勃发明了一种神奇的排序算法：按位排序。用该算法对一个序列进行排序之后，拆出这个序列二进制表示的每一位来看，都是有序的（升序或者降序都可以）。

聪明的你很快意识到，并不是所有的序列通过调换顺序都能达到二进制每一位都是有序的效果。刚刚学习了编程知识的你决定将鲍勃的算法用代码实现出来，并向他证明有的序列是可以通过这种排序算法产生结果，但是有些序列是不能产生正确结果的。

鲍勃正准备公开发表他的错误算法，请你赶快阻止他吧！

Input

第一行输入一个正整数 T ($1 \leq T \leq 10$)，表示有 T 组数据。
对于每组数据，第一行一个整数 n ($1 \leq n \leq 2 \times 10^5$)，代表原始序列的长度
其后一行 n 个整数 a_1, a_2, \dots, a_n ($0 \leq a_i < 2^{30}$)

Output

对于每组数据，输出一行：
如果无解，输出”No Solution”(不包括引号)。
如果有解，输出一行 n 个整数，代表排序后的结果。如果有多种解，请你输出字典序最小的那一种。
两个序列的字典序比较方式是从左到右找到第一个不同的元素，比较该元素的大小。

Example

bitsort.in	bitsort.out
2	3 2 6 6 4
5	No Solution
2 3 4 6 6	
3	
3 5 6	

Scoring

测试点编号	$n \leq$	特殊性质
1	10	无
2 ~ 4	1000	$a_i \leq 2^{10} - 1$
5 ~ 7	2×10^5	保证有解
8 ~ 10	2×10^5	无

Problem B. 鲍勃的数学题

Input file: `math.in`
Output file: `math.out`
Time limit: 3 second
Memory limit: 256 megabytes

继上次鲍勃在排序算法的研究上遭遇失败之后，这次他开始研究数论问题。他学习了欧拉函数的定义： $\phi(n)$ 等于小于等于 n 的与 n 互质的正整数的个数。

鲍勃又发现了一种神奇的数字，这些数字 x 满足 $\phi(x)$ 的最大质因子不超过 5。

鲍勃想让你帮忙计算一下， $1 \sim n$ 内神奇数字的和是多少？由于答案过大，请你输出答案对 2^{32} 取模后的值

Input

第一行一个整数 $n(1 \leq n \leq 10^{12})$ ，代表题目中的范围。

Output

对于每组数据，输出一行一个答案，代表 $1 \sim n$ 内神奇数字的和。对 2^{32} 取模后的值

Example

<code>math.in</code>	<code>math.out</code>
10000000000000	939087315
23	253

Scoring

共 10 组数据	测试点	$n \leq$
	第 i 组数据	$10^{(i+2)}$

Note

第一个不神奇的数字是 23， $\phi(23) = 22 = 2 * 11$

Problem C. 最大子段和

Input file: subarray.in
Output file: subarray.out
Time limit: 3 second
Memory limit: 512 megabytes

鲍勃是一个顽强的少年，继排序研究和数学研究遭遇困难之后，他意识到算法的研究不能停留在理论层面，必须也有扎实的编程技巧作为基础。于是他决定学习编程的知识。

鲍勃了解到，在编程中动态规划是非常重要的一个知识点。在他研究动态规划的时候遇到了这样一个问题：“最大子段和问题”，这个问题是：给你一个序列，你需要找出其中的一个子段（不一定非空）使这一段的和是序列的所有段中最大的。

鲍勃突发奇想：这个题能不能改成：多次询问每次给一个区间，找出来这个区间内的最大子段和？

可惜的是，这似乎不是一个容易的问题，鲍勃又来寻求你的帮助了。

Input

第一行一个 $n(1 \leq n \leq 2 \times 10^6)$ ，代表序列的长度

其后一行 n 个数字，代表原序列 A ，下标从 $1 \sim n$ ， $|A_i| \leq 10^3$

其后一行一个 $q(1 \leq q \leq 10^7)$ ，代表询问的个数

其后输入一行两个整数 $k1, k2(0 \leq k1, k2 \leq 2^{32} - 1)$ ，其作用下方会说明

由于本题的询问次数过多，为了避免 IO 时间影响算法效率，本题采用特殊的方法来读入询问。

将下方代码中的 $k1, k2$ 设为输入的 $k1, k2$

```
unsigned k1, k2;

unsigned long long xorShift128Plus() {

    unsigned long long k3 = k1, k4 = k2;

    k1 = k4;

    k3 ^= k3 << 23;

    k2 = k3 ^ k4 ^ (k3 >> 17) ^ (k4 >> 26);

    return k2 + k4;

}
```

请你用下方的方式生成询问的区间

```
for(int i=1;i<=m;i++){  
  
    l[i]=xorShift128Plus()%n+1;  
  
    r[i]=xorShift128Plus()%n+1;  
  
    if(l[i]>r[i]) swap(l[i],r[i]);  
  
}
```

Output

为了避免输出数据过多影响算法速度，你只需要输出一行一个整数代表每次询问的答案的异或和。

Example

subarray.in	subarray.out
5 1 -2 3 -1 5 2 998244353 1000000007	7

Note

样例中，998244353 1000000007 作为 k1,k2, 当 n 等于 5 时产生的前两个区间为 [2,2],[3,5]。这两个区间的最大子段和分别为 0 和 7，所以异或和是 7

Scoring

测试点编号	$n \leq$	$q \leq$
1 ~ 3	1000	1000
4 ~ 7	2×10^5	2×10^5
8 ~ 10	2×10^6	10^7

Problem D. 鲍勃打比赛

Input file: contest.in
Output file: contest.out
Time limit: 4 second
Memory limit: 256 megabytes

终于，鲍勃的算法和编程水平都有了不小的进步，他决定参加 noip 来测试自己的水平。为了练习，爱丽丝老师为他准备了 n 场训练赛，第 i 场的难度是 a_i ，鲍勃可以从第 1 场开始，依次考虑参加不参加。他参加比赛的规则如下：

- 如果鲍勃同时参加了比赛 i, j ($i < j$) 则应该有 $a_i \leq a_j$
- 如果鲍勃没有参加比赛 i ，并且这是连续第 k 场没有参加的比赛，他将丢失 k 点能力值。
- 鲍勃最终获得的能力值是所有参加的比赛的难度和减去丢失的能力值的和。
- 爱丽丝老师想让你帮忙计算一下，鲍勃能获得的最大的能力值是多少？

Input

- 输入第一行一个 $T(1 \leq T \leq 5)$ 代表数据组数。
- 对于每组数据：
 - 第一行一个整数 $n(1 \leq n \leq 10^5)$ ，代表比赛的个数
 - 其后一行 n 个整数 $a_i, (-10^9 \leq a_i \leq 10^9)$ ，代表第 i 个比赛的困难程度。

Output

对于每组数据，输出一行一个整数，代表鲍勃能获得的最大的能力值

Example

contest.in	contest.out
2	7
7	-11
1 3 2 7 3 2 4	
7	
-3 -4 -2 -2 -6 -8 -1	

Scoring

测试点编号	$n \leq$	$ a_i \leq$
1 ~ 2	1000	10^9
3 ~ 5	10^5	10^3
6 ~ 7	10^5	a_i 不为负
8 ~ 10	10^5	10^9