

补题笔记 · CodeForces#735(Div.2)

唐冬 \LaTeX

更新: August 12, 2021

1 A.Cherry

1.1 题意

原题[链接](#)

给出一个正整数数列，找出一个连续子序列，使得序列中最大值与最小值乘积最大。

1.2 思路

这次 cf 比赛感觉就像是数学专场，看懂题目相对容易，代码也比较短，但想出题解的难度简直没谁了，下次一定好好学数论 QAQ。

这个题表面上说是找区间，但通过观察可以发现，只有区间内最大值和最小值影响结果，和其他值无关。

现在，我们想办法找出两个最值。易证：正整数序列相邻两数乘积为任意区间内最大最小值乘积的最大值。

写易证会被扣分 (?)，那我尝试证明一下。首先在曲线上选两个连续的点，构成一个小区间，最大值为 A 点，最小值为 B 点。

我们在左边取一点 C 构成新区间 [C,A]，发现最小值变小了；如果在右边取一点 D 构成区间 [B,D]，发现最大值变大，但最小值却不是最佳（向右移后变大）。所以，遍历整个数组，找出相邻两数乘积的最大值就是题解。

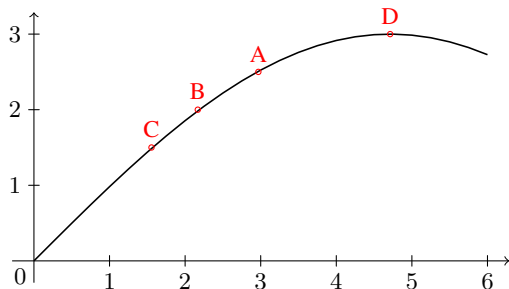


图 1

1.3 代码

```
1 #include "bits/stdc++.h"
2
3 using namespace std;
4
5 const int nmax=1e5+10;
6
7 long long val[nmax],l_val[nmax],r_val[nmax];
8
9 int main(){
10     freopen("input.txt","r",stdin);
11     int T;
12     cin>>T;
13     while(T--){
14         int n;
15         cin>>n;
16         for(int i=0;i<n;i++){
```

```

17         cin>>val[i];
18     }
19     long long ans=-1,max_l=val[0];
20     for(int i=1;i<=n;i++){
21         max_l=max(max_l,val[i-1]);
22         memcpy(r_val,val,sizeof(val));
23         sort(l_val,l_val+i);
24         sort(r_val+i,r_val+n);
25         ans=max(l_val[i-1]*r_val[i],ans);
26     }
27     cout<<ans<<endl;
28 }
29 }

```

2 B. Cobb

2.1 题意

原题[链接](#)

给出数列 $a_1, a_2, a_3, \dots, a_n$ ，求全部数对 (a_i, a_j) 中 $i \cdot j - k \cdot (a_i | a_j)$ 的最大值 ($|$ 为 **or** 运算)。

数据范围： $2 \leq n \leq 10^5$ ， $1 \leq k \leq \min(n, 100)$ 。

2.2 思路

全 遍历时间复杂度 $\mathcal{O}(n^2)$ 会超时，我们考虑优化。设

$$f(i, j) = i \cdot j - k \cdot (a_i | a_j), i < j \quad (1)$$

由于 i, j 的范围, $(i|j)$ 的最大值是小于等于 $2n$ 的, 只考虑 $i \cdot j$ 部分, 它的最大值等于 $n \cdot (n - 1)$, 所以, 最后两个数的最小值可能是:

$$\begin{aligned}f_{\min}(n - 1, n) &= n(n - 1) - 2kn \\&= n^2 - 2kn - n\end{aligned}\tag{2}$$

任意数对的最大值为:

$$f_{\max}(i, j) = i \cdot j\tag{3}$$

由公式 (3) 得:

$$f_{\max}(i, n) = i \cdot n\tag{4}$$

令 $f(i, n) > f(n - 1, n)$, 则:

$$\begin{aligned}f_{\max}(i, n) &> f_{\min}(n - 1, n) \\i \cdot n &> n^2 - 2kn - n \\i &> n - 2k - 1\end{aligned}\tag{5}$$

从上面的推导可以得出, 要想得出最大值, i, j 必须大于 $n - 2k - 1$, 也就是说, 只需分别遍历 $2k$ 次就能求出答案, 时间复杂度变为 $\mathcal{O}(k^2)$, 而题目中 k 的数据范围远小于 n , 所以能在规定时间内完成遍历。

2.3 代码

```
1 #include "bits/stdc++.h"
2
3 g namespace std;
4
5 t int nmax=1e5+10;
6 a[nmax];
7
8 ine ll long long
```

```

9
10 main(){
11     freopen("input.txt", "r", stdin);
12     int T;
13     cin>>T;
14     while(T--){
15         int n,k;
16         cin>>n>>k;
17         for(int i=0;i<n;i++) cin>>a[i];
18         ll ans=-1e12;
19
20         ll Fst=max(0,n-2*k);
21         for(int i=Fst;i<n;i++) for(int j=i+1;j<n;j++){
22             ans=max(ans, 1ll*(i+1)*(j+1)-1ll*k*(a[i]|a[j]));
23         }
24         cout<<ans<<endl;
25     }

```

3 C.Mikasa

3.1 题意

原题[链接](#)

给出两个数 m, n ，求数列 $n \oplus 1, n \oplus 2, \dots, n \oplus m$ 的 **MEX** 值。

规定数列的 **MEX** 为该数列与自然数列补集的最小值，例如 $\text{MEX}(0,1,2,4) = 3$ ， $\text{MEX}(1,2021) = 0$ 。

3.2 思路

若 $a \oplus b = c$ ，则 $a \oplus c = b$ 。

所以只要从 0 开始递增遍历所有的 c ，当第一个大于 m 的 c 出现时，就是答案，时间复杂度 $\mathcal{O}(n)$ 。

但是题目中 n 和 m 实在是太大（高达 10^9 ），所以我们还需要继续优化。

由于异或运算能够改变指定二进制位的值，所以我们位运算的角度考虑如何求 c 的最小值：

- 若 n 的二进制位数多于 m （即 $n > m$ ），则答案为 0。
- 若 n 的二进制位数少于或等于 m ，要使 $n \oplus c$ 大于 m ，则需要将 n 的二进制 0 位从高到低逐个变成 1 位，直到其大于 m 为止。此时参与异或运算的 c 即为最小值。 c 的值为变零位是 1，其他位补零。

例： $n = 101010_{(2)}$, $m = 110101_{(2)}$, $n \oplus c = 111010_{(2)}$, $c = 10000_{(2)} = 16_{(10)}$

此时，时间复杂度为 $\mathcal{O}(\log(n))$ 。

水平叭太行感觉写得好绕 QAQ，可以参考原题解，[链接](#)。

3.3 代码

```
1  #include "bits/stdc++.h"
2
3  g namespace std;
4
5  main(){
6  freopen("input.txt", "r", stdin);
7  int t;
8  cin >> t;
9  while(t--){
10     int n, m;
11     cin >> n >> m;
12     m++;
```

```

13     int ans=0;
14     for(int i=30;i>=0&& n<m;i--) if((n>>i&1) != (m>>i&1)){
15         ans|=1<<i;
16         n|=1<<i;
17     }
18     cout<<ans<<endl;
19 }

```

4 D.Diane

4.1 题意

原 题链接

构造一个长度为 n 的字符串，使得它的所有非空子字符串的出现次数为奇数次。

4.2 思路

首先要正确理解题意，题目中是子字符串的出现次数，不是子字符串中字母出现次数（我一开始没看懂题，加上具有诱导的样例，就再一次泪目了 QAQ），此题题解极为精妙，看完后我直呼“妙啊”。

首先考虑一个字母的情况。在字符串 $\underbrace{aaa\dots a}_n$ 中，“a”出现了 n 次，“aa”出现了 $n-1$ 次，“aaa”出现了 $n-2$ 次，以此类推，若 n 为奇数，则出现次数依次是奇数，偶数，奇数……

由于奇数 + 偶数 = 奇数，所以在构造数列时，设 $k = \lfloor \frac{n}{2} \rfloor$ ：

· 若 n 为偶数，则构造数列为 $\underbrace{a\dots a}_{k-1} b \underbrace{a\dots a}_k$ ，前半部分“a”, “aa”, “aaa”... 出现次数为偶，奇，偶，奇...，后半部分对应字符串出现次数为奇，偶，奇，偶...，相加后的总次数均为奇数次。

· 若 n 为奇数，则构造数列为 $\underbrace{a \dots a}_{k-1} b c \underbrace{a \dots a}_{k-2}$ ，原理同上。

按上述方法构造即为答案。

妙啊~~~~

在验证答案时，由于子串数量巨大，oj 用了个后缀自动机感兴趣的同学可以去原题解看看，我并没有看懂 QAQ。

4.3 代码

```
1  #include "bits/stdc++.h"
2
3  namespace std;
4
5  main(){
6  freopen("input.txt", "r", stdin);
7  int T;
8  cin>>T;
9  while(T--){
10     int n;
11     cin>>n;
12     int k;
13     if(n==1) cout<<'a'<<endl;
14     else if(n%2==0){
15         k=(n-1)/2;
16         for(int i=0;i<k;i++) cout<<'a';
17         cout<<'b';
18         for(int i=0;i<k+1;i++) cout<<'a';
19         cout<<endl;
20     }
21     else{
22         k=(n-2)/2;
23         for(int i=0;i<k;i++) cout<<'a';
```



```
24         cout<<"bc";
25         for(int i=0;i<k+1;i++) cout<<'a';
26         cout<<endl;
27     }
28 }
```

5 E.You

5.1 题意

原 题链接

给出一棵树，按不同的顺序依次删除其每一个节点，记录删除时该节点连接的其他节点的数量，从节点 1 到 n 组成数列，求该数列的最大公约数并对 998244353 取模，统计不同删除顺序下产生的数列的最大公约数相同的个数。

5.2 思路

很 难，我还没有完全看懂，决定先去玩一会儿稻妻，待会儿再写-v-