

Codeforces Round #768 (Div.2) A~C 题解

A. Min Max Swap

题意

共 t ($1 \leq t \leq 100$) 组测试数据，每组中给出一个数组的元素个数 n ($1 \leq n \leq 100$)，接下来两行分别是 n 个元素，分别为两个数组。可以将两个数组对应位置的元素交换，使得两个数组中最大值的乘积最小，求出最小乘积。

思路

注意读题不要读错了，不然会浪费很多时间。

找出所有元素中最大的元素，它是一定会被乘到的，我们只需让另外一个最大值最小即可，所以我们应该使大的数尽量往最大元素所在组放，随后找出另一组的最大值即可。

也就是分别求出对应位置比较后的较大值的最大值和较小值的最大值。

总结

一开始心急没仔细看，以为是可以将两个数组中的任意元素互换，浪费了不少时间。以后要好好审题。

AC代码

Problem	Lang	Verdict	Time	Memory
A - Min Max Swap	GNU C++17	Accepted	15 ms	0 KB

```
1 #include<bits/stdc++.h>
2 using namespace std;
3 int n;
4 int a[110];
5 int b[110];
6 void solve()
7 {
8     cin >> n;
9     for (int i = 1; i <= n; i++)
10         cin >> a[i];
11     for (int i = 1; i <= n; i++)
12         cin >> b[i];
13     int Max1 = 0;
14     int Max2 = 0;
15     for (int i = 1; i <= n; i++) {
16         Max1 = max(Max1, min(a[i], b[i]));
17         Max2 = max(Max2, max(a[i], b[i]));
18     }
19     cout << Max1 * Max2 << endl;
20 }
21
22 int main()
23 {
```

```

24     ios::sync_with_stdio(false);
25     cin.tie(0); cout.tie(0);
26     int tt;
27     cin >> tt;
28     while (tt--) {
29         solve();
30     }
31 }

```

B. Fun with Even Subarrays

题意

t ($1 \leq t \leq 2 \cdot 10^4$)组元素, n ($1 \leq n \leq 2 \cdot 10^5$)个元素, 可进行的操作如下:

选定一个区间 $[l, r]$, 将 $[r+1, r+r-l]$ 区间内的元素对应复制给区间 $[l, r]$, 显然这两个区间都需在总区间内。

求出使数组中所有元素相等的最小操作数。

思路

因为它是一个将右边的数复制到左边的操作, 所以应该从右边开始操作。如果遇到不同的数, 就进行一次操作覆盖, 遇到相同的就可以继续推进, 操作新覆盖的长度是等于它到起点的长度。

总结

遇到runtime error的报错应该去检查是否存在数组越界、爆栈。

本题我当时写的是将跳转后的点的值改变成右端点, 再来进行下一次比较的。那么便出现了一个问题, 最后一次跳转是可能到负下标的, 那么对数组负下标元素的赋值在我本地是没问题的, 但在线上测试时就runtime errors了。

AC代码

Problem	Lang	Verdict	Time	Memory
B - Fun with Even Subarrays	GNU C++17	Accepted	108 ms	800 KB

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  int n;
4  int a[200010];
5  void solve()
6  {
7      cin >> n;
8      for (int i = 1; i <= n; i++)
9          cin >> a[i];
10     int k = n;
11     int cnt = 0;
12     int j = 1;
13     while (k>1) {

```

```

14         if (a[k] == a[k - 1]) k--;
15         else {
16             int p = k;
17             k -= j;
18             if(k>0) a[k] = a[p];
19             //注意下标不能为负，不然会runtime error
20             cnt++;
21         }
22         j = n - k + 1;
23     }
24     cout << cnt << endl;
25 }
26
27 int main()
28 {
29     ios::sync_with_stdio(false);
30     cin.tie(0); cout.tie(0);
31     int tt;
32     cin >> tt;
33     while (tt--) {
34         solve();
35     }
36 }

```

C. And Matching

题意

t ($1 \leq t \leq 400$)组测试，每组给出一个 n 和 k ， $4 \leq n \leq 2^{16}$, $0 \leq k \leq n - 1$, $n = 2^j, j \in \mathbb{Z}$

对于 0 到 $n - 1$ 的这些数，将其两两分组，使得各组的两个数 `a&b` 的总和等于 k ，输出任意一种分组方式，若不行，输出-1。

思路

数据很大，不能暴力枚举，那显然就是一道构造题。

分类讨论，先看一种比较特殊的情况，即 $n = 0$ 时，需要使得所有 `a&b` 均为 0。

设 x 的组员应为 $c(x)$ ，那么怎么求 $c(x)$ 呢？

由于同一为一，不然则为0，所以不同的时候一定是0那么就可以将 x 按位取反，求得 $c(x)$

又可以发现由于 $n = 2^j$ ， $n - 1$ 转化成二进制便全由1组成，令 `x^(n-1)`，即可使得 x 按位取反，求得 $c(x)$

再分析剩下两种情况。

当 $0 < k < n - 1$ 时，使 0 和 $c(k)$ 配对， k 和 $n - 1$ 配对即可。

当 $k = n - 1$ 时，若 $n = 4$ 显然输出 -1；其余情况，使 0 和 2 配， $n - 1$ 和 $n - 2$ 配， $n - 3$ 和 1 配即可。

总结

对于构造类题目，可以先尝试思考特殊情况，说不定能推广到一般。

AC代码

Problem	Lang	Verdict	Time	Memory
C - And Matching	GNU C++17	Accepted	108 ms	0 KB

```
1  #include<bits/stdc++.h>
2  using namespace std;
3  int n;
4
5  int c(int x)
6  {
7      return x ^ (n - 1);
8  }
9
10 void solve()
11 {
12     int k;
13     cin >> n >> k;
14     if (k == 0) {
15         for (int i = 0; i <= n / 2 - 1; i++)
16             cout << i << ' ' << c(i) << endl;
17     }
18     else if (k == n - 1) {
19         if (n == 4) {
20             cout << -1 << endl;
21         }
22         else {
23             cout << n - 1 << ' ' << n - 2 << endl;
24             cout << n - 3 << ' ' << 1 << endl;
25             cout << 0 << ' ' << 2 << endl;
26             for (int i = 3; i <= n / 2 - 1; i++)
27                 cout << i << ' ' << c(i) << endl;
28         }
29     }
30     else {
31         cout << 0 << ' ' << c(k) << endl;
32         cout << k << ' ' << n - 1 << endl;
33         for (int i = 1; i <= n / 2 - 1; i++)
34             if(i!=k&&i!=c(k))cout << i << ' ' << c(i) << endl;
35     }
36 }
37
38 int main()
39 {
40     int tt;
41     cin >> tt;
42     while (tt--) {
43         solve();
44     }
45     return 0;
```

by syj

希望哪天div2能在场上出一道c啊。