2020.07.23codeforces加训

比赛情况

题号	A	B	C	D	E	F	G	H	I	J	K
状态	O	-	-	-	-	O	O	!	O	O	O

O 在比赛中通过 Ø 赛后通过 ! 尝试了但是失败了 - 没有尝试

比赛时间

2020-07-23 12:00-17:00

题解

F - Empty Vessels

可以把最大的桶作为中介构造目标，之后的东西就是搜索并记录路径就好了。

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
const int N = 2e4+5;
int a[15],maxi;
bool vis[N];
int from[N];
queue<int>Q;
void print(int x,int dep) {
    if (x==0) {
        printf("%d\n",dep);
        return ;
    }
    if (a[from[x]]<=x) {
        print(x-a[from[x]],dep+2);
        printf("1 %d\n3 %d %d\n",from[x],from[x],maxi);
    } else {
        print(x-a[from[x]]+a[maxi],dep+4);
        printf("1 %d\n3 %d %d\n2 %d\n3 %d
%d\n",from[x],from[x],maxi,maxi,from[x],maxi);
    }
}
int main()
{
    int n,A;
    scanf("%d%d",&n,&A);
    maxi = 1;
```

```
    for (int i = 1;i<= n;i++)
    {
        scanf("%d",&a[i]);
        if (a[i]>a[maxi])maxi = i;
    }
    vis[0] = true;
    Q.push(0);
    while (!Q.empty()) {
        int x = Q.front();
        Q.pop();
        for (int i = 1;i<= n;i++)
            if (!vis[(x+a[i])%a[maxi]]) {
                vis[(x+a[i])%a[maxi]] = true;
                from[(x+a[i])%a[maxi]] = i;
                Q.push((x+a[i])%a[maxi]);
            }
    }
    if (!vis[A]) {
        if (A==a[maxi]) printf("1\n1 %d\n",maxi);
        else printf("-1");
        return 0;
    }
    print(A,0);
    return 0;
}
```

G - Maximum Product

可以意识到，乘积最大的必然是在区间L,R内并且有若干个9的，我们考虑找一个小于R的，最后若干位为9的数字，然后看这个数字是否大于L□如果大于则可以更新答案。

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
ll calc(ll a) {
    ll ans = 1;
    while (a) {
        ans = ans*(a%10);
        a = a/10;
    }
    return ans;
}
ll Pow10(int x) {
    ll ans = 1;
    for (int i = 1;i<= x;i++)
        ans = ans*10;
```

```
    return ans;
}
int main()
{
    ll a,b;
    scanf("%lld%lld",&a,&b);
    ll ans = calc(b);
    ll tans = b;
    for (int i = 1;i<= 18;i++) {
        ll now = b;
        int newn[20]={0},newlen=0;
        for (int j = 1;j<= i && now;j++) {
            if (now >= 10) {
                if (now%10!=9) now -= 10;
                newn[++newlen] = 9;
            } else {
                newn[++newlen] = now;
            }
            now /= 10;
        }
        ll tnewn = 0;
        for (int i = newlen;i>=1;i--)
            tnewn = tnewn*10+newn[i];
        tnewn = now*Pow10(newlen)+tnewn;
        if (tnewn >= a && calc(tnewn) > ans) {
            ans = calc(tnewn);
            tans = tnewn;
        }
    }
    printf("%lld\n",tans);
    return 0;
}
```

比赛总结与反思