codeforces round 644

A

题意：给两个长方形边长a和b□问最小要多大正方形放这两个长方形。

题解：经过简单枚举□$2*min(a,b)$和$max(a,b)$取一个最小值即为边长。略代码。

B

题意：一个集合分成两个集合，问第一个集合的最大值和第二个集合的最小值的差的绝对值最小能是多少。

题解：排序，然后寻找相隔最近的两个数。略代码。

C

题意：定义合法的一对数为这些数都是同奇偶的，或者两数相差1，给定集合容量n□n为偶数，问能否使一个集合中所有数都分成合法的对？

题解：计算集合中偶数和奇数的个数，若偶数的个数为奇数，则暴力找能否找到一对满足俩数相差1，找不到就不能。代码略。

D

题意：定义第i种包装有i个一样的商品，一共只有k种，问要集齐n个商品，且只能购买一种包装，问最少需要卖多少个。

题解：即求n小于等于k的最大因子，只需要$O(\sqrt n)$遍历即可。代码略。

E

题意：题意不怎么好表达，放个链接把……，Polygon

题解：由1产生的特征，不难发现若非最后一行且非最右边一行出现1，则必有该点右边或者下边必定至少有一个1，因为1不可能凭空产生。将其作为合法的条件即可。代码略。

F

题意：给定n个字符串，求一个字符串，使该字符串满足，与任意一个字符串都有按位不同的字符不超过1个。

题解：这题居然卡了1个小时，~~我不做人啦~~，范围很小，以第一个字符串为基准，枚举即可，每次改变第一个字符串的一个位置的字符。

```
#include <bits/stdc++.h>
using namespace std;
char s[500][500];
char ans[500];
int t,n,m;
bool check()
{
    for(int i=1;i<=n;i++)
    {
        int cnt=0;
        for(int j=1;j<=m;j++)
        {
            if(s[i][j]!=ans[j])
            {
                cnt++;
                if(cnt>=2) return false;
            }
        }
    }
    return true;
}
int main()
{
    scanf("%d",&t);
    while(t--)
    {
        int yes=0;
        scanf("%d%d",&n,&m);
        for(int i=1;i<=n;i++)
        {
            scanf("%s",s[i]+1);
        }

        for(int i=1;i<=m;i++)
        {
            for(int j=1;j<=m;j++) ans[j]=s[1][j];
            for(int j='a';j<='z';j++)
            {
                ans[i]=j;
                if(check())
                {
                    yes=1;
                    break;
                }
            }
        if(yes) {
            break;}
        }
        if(yes)
```

```
        {
            for(int i=1;i<=m;i++) printf("%c",ans[i]);
            puts("");
        }
        else printf("-1\n");
    }
}
```

G

题意：给定义1个$n*m$的矩阵，初始全为0，问有没有每一种填充1的方案，能满足每一行都有a个1，且每一列都有b个1。

题解：要满足根据数量关系一定有$m*a=n*b$值就可以用作判断成不成立的条件。若成立，考虑一种构造方案，从第1列开始，每次在向右填充1，若填满了则从改行的第一个开始填充起，每填满a个则换行，这样构造出来的矩阵一定是合法的(~~别问为啥，问就是观察法~~)

```
#include <bits/stdc++.h>
using namespace std;
char s[100][100];
int main()
{
    int t,n,m,a,b;
    scanf("%d",&t);
    while(t--)
    {
        memset(s,'0',sizeof(s));
        scanf("%d%d%d%d",&n,&m,&a,&b);
        if(n*a!=m*b)
        {
            printf("NO\n");
            continue;
        }
        for(int j=0,i=0;i<n;i++)
        {
            for(int k=1;k<=a;j++,k++)
            {
                s[i][j%m]='1';
            }
        }
        printf("YES\n");
        for(int i=0;i<n;i++)
        {
            for(int j=0;j<m;j++) printf("%c",s[i][j]);
            puts("");
        }
    }
```

```
}
```

H

题意：定义长度为m的01字符串，其按照字典序排序，去除给定的n个字符串后，问最中间的字符串(即第$\lfloor\frac{n-1}{2}\rfloor$位)是哪一个。

题解：观察字符串由01组成，立刻就能想到2进制，可以考虑把所给数全部转化为十进制数，问题即转化为求一系列数中位数问题，可以二分去做，一定要注意边界条件(每次check小于等于目标的数，只是检查小于目标的数会wa.....)要不然会死也调不出来……。然后最后注意要转化为2进制输出，缺0补齐。

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
ll a[400];
int n,t,m;
char s[400];
int ans[400];
bool check(ll x) { //检查是否为中位数
    ll res=x+1;//小于等于x的数(为了去除等于x的数)
    for(int i=1; i<=n; i++) {
        if(a[i]<=x) {
            res--;
        }
    }
    return res>=((1ll<<m)-1-n)/2+1;
}
int main() {
    scanf("%d",&t);
    while(t--) {
        scanf("%d%d",&n,&m);
        memset(a,0,sizeof(a));
        memset(ans,0,sizeof(ans));
        for(int i=1; i<=n; i++) {
            scanf("%s",s);
            for(int j=0; s[j]; j++) a[i]=1ll*a[i]*2+s[j]-'0';
        }
        ll l=0,r=1ll<<m;
        while(l<r) {
            ll mid=(l+r)>>1;
            if(check(mid)) r=mid;
            else l=mid+1;
        }
        int cnt=0;
        while(l) {
            ans[++cnt]=l%2;
            l/=2;
```

```
        }
        for(int i=m; i>=1; i--) printf("%d",ans[i]);
        puts("");
    }
}
```