poj3970

最大公约数$lcm(a,b)=\frac{a*b}{gcd(a,b)}$

```
#include<iostream>
using namespace std;
long long int gcd(long long int ans ,long long int x){
    if(ans%x==0){
        return x;
    }else{
        return gcd(x,ans%x);
    }
    return 0;
}
int main() {
    int n=0;
    long long ans=0;
    long long x=0;
    while(scanf("%d",&n)==1&&n){
        ans=0;
        x=0;
        //scanf("%lld",ans);
        cin>>ans;
        --n;

        while(n--){
            scanf("%lld",&x);
            ans=ans*x/(gcd(ans,x));
        }

        if(ans>=1000000){
            printf("Too much money to pay!\n");
        }else {
            printf("The CEO must bring %lld pounds.\n",ans);
        }

    }
    return 0;
}
```

poj1995

快速幂 霍纳法则

$Pn(x)= a_n{x_n} +a_{n−1}{x_n−1}+…+a_1x+a *0$ $=((…(((a_nx +a*{n−1})x+a_{n−2})x+ a{n−3})…)x+a1)x+ a_0$

```
#include <iostream>
#include <stdio.h>
using namespace std;
long long int mi(long long int a,long long int b,long long int m){
    long long int an=1ll;
    a%=m;
    while(b){
        if(b&1) an=(an*a)%m;
        a=(a*a)%m;
        b>>=1;
    }
    return an;
}
int main(){
    int z=0;
    long long int m=0;
    int n=0;
    long long ans=0;
    long long a=0,b=0;
    scanf("%d",&z);
    while(z--){
        ans=0;
        scanf("%lld",&m);
        scanf("%d",&n);
        while(n--){
            scanf("%lld %lld",&a,&b);
            //cout<<ans<<endl;
            ans=(ans+mi(a,b,m))%m;
        }
        //ans%=m;
        printf("%lld\n",ans);

    }
    return 0;
}
```

poj3070

```
Source Code
#include <iostream>
#define mod 10000
using namespace std;
typedef long long ll;
typedef struct matrix{
    int elm11;
    int elm12;
    int elm21;
    int elm22;
};
matrix matrixmult(matrix m,matrix b){
    matrix ans;
    ans.elm11=(m.elm11*b.elm11+m.elm12*b.elm21)%mod;
    ans.elm12=(m.elm11*b.elm12+m.elm12*b.elm22)%mod;
    ans.elm21=(m.elm21*b.elm11+m.elm22*b.elm21)%mod;
    ans.elm22=(m.elm21*b.elm12+m.elm22*b.elm22)%mod;
    return ans;

}
matrix Fib(matrix m,ll n){
    if(n==1)  return m;
    else if(n&1) return matrixmult(Fib(m,n-1),m);
    else {
        matrix tmp = Fib(m,n/2);
        return matrixmult(tmp,tmp);
    }
}
int main(){
    ll n=0;
    matrix ans;
    matrix matrix1;
    matrix1.elm11=1;
    matrix1.elm12=1;
    matrix1.elm21=1;
    matrix1.elm22=0;
    //cout<<"sd"<<endl;
    while(scanf("%lld",&n)==1&&n!=-1){

        if(!n){
            printf("0\n");
            continue;
        }
        ans=Fib(matrix1,n);
        printf("%d\n",ans.elm21);
    }
    return 0;
}
```

poj3233

快速幂+二分，观察得：

当k为偶数时：

$S(k)=(1+A^\frac{k}{2})*(A+A^2+...+A^{\frac{k}{2}})$ $ =(1+A^\frac{k}{2})*S(\frac{k}{2})$

当k为奇数时：

$S(k)=(1+A^\frac{k-1}{2})*S(\frac{k}{2})+A^k$

```
#include <iostream>
#include <cstring>
using namespace std;
typedef long long ll;
int n,mod;
struct matrix{
    int matr[40][40];
}a;
matrix mat_mult(matrix a,matrix mat,int n,int mod){
    matrix ans;

    memset(ans.matr, 0, sizeof(ans.matr));

    for(int i = 0; i < n; i++)
        for(int j = 0; j < n; j++)
            if(a.matr[i][j])
                for(int k = 0; k < n; k++)
                    ans.matr[i][k] = (ans.matr[i][k] + a.matr[i][j] * mat.matr[j][k])
% mod;
    return ans;
}
matrix mat_addone(int n,matrix mat){
    for(int i=0;i<n;i++)
        mat.matr[i][i]=mat.matr[i][i]+1;
    return mat;
}
matrix mat_add(matrix a,matrix b,int n,int mod){
    matrix ans;
    memset(ans.matr, 0, sizeof(ans.matr));
    for(int i=0;i<n;i++)
        for(int j=0;j<n;j++){
            ans.matr[i][j]=(a.matr[i][j]+b.matr[i][j])%mod;
        }
    return ans;
}
matrix calucutea(matrix a,int k,int n,int mod){
    matrix ans;
    memset(ans.matr, 0, sizeof(ans.matr));

    for(int i = 0; i < n; i++)
        ans.matr[i][i] = 1;

    while(k)
    {
        if(k & 1)
            ans = mat_mult(ans, a,n,mod);
        k >>= 1;
        a = mat_mult(a,a,n,mod);
    }

    return ans;
}
matrix rec(int n,int mod,matrix a,ll k){
    if(k == 1)
        return a;

    matrix ans;
    memset(ans.matr, 0, sizeof(ans.matr));

    for(int i = 0; i < n; i++)
        ans.matr[i][i] = 1;
    ans = mat_add(ans, calucutea(a, k >> 1,n,mod),n,mod);
    ans = mat_mult(ans, rec(n,mod,a, k >> 1),n,mod);
    if(k & 1) //奇数
        ans = mat_add(ans, calucutea(a,k,n,mod),n,mod);
    return ans;
}

int main() {
    matrix a,ans;
    ll k = 0;
    int tmp = 0;
    scanf("%d %lld %d", &n, &k, &mod);
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++) {
            scanf("%d", &tmp);
            a.matr[i][j] = tmp % mod;
        }

    ans=rec(n,mod,a,k);

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            printf("%d ", ans.matr[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

poj2092

实现计数排序

```
#include<stdio.h>
#include<string.h>
#include<algorithm>
using namespace std;
const int maxn = 10000+5;
int a[maxn];
struct Node{
    int num;
    int index;
}node[maxn];
//声明结构体数组
bool cmp(Node a, Node b)
{
    if(a.num == b.num) return a.index < b.index;
    else return a.num > b.num;
}
//定义排序顺序
int main()
{
    int n,m;
    while(scanf("%d%d", &n,&m) != 0)
    {
        if(n == 0 && m == 0) break;
        int x;
        int Max = 0;

        for(int i = 0; i < maxn; i++)
        {
            a[i] = 0;
            node[i].num = 0;
            node[i].index = 0;
        }
       //清空结构体数组
        int N = n*m;
        //总数为n*m
        for(int i = 0; i < N; i++)
        {
            scanf("%d", &x);
            a[x]++;
            Max = max(Max, x);
        }
        N = Max+1;
        //找到最大的数
        int j = 0;
        for(int i = 1; i < N; i++)
        {
            if(a[i] != 0)
            {
                node[j].num = a[i];
                node[j].index = i;
                j++;
            }
        }
        //将有值的位拷贝到结构体数组中
        sort(node,node+j,cmp);
        //对数组按出现次数排序
        N = j;
        //记录个数
        int s = node[0].num;
        //最大值
        int index = 0;
        for(int i = 0; i < N; i++)
        {
            if(node[i].num < s)
            {
                s = node[i].num;
                index = i;
                break;
            }
        }
//求最后一个最大的元素好二大的值
        for(int i = index; i < N; i++)
        {
            if(node[i].num == s)
            {
                printf("%d ", node[i].index);
            }
            else if(node[i].num > s) break;
        }
        printf("\n");
    }
    return 0;
}
```

poj2053

英文词典查词,第一遍没过竟然用了AVL,挠头

```
#include<cstdio>
#include<cstring>
#include<iostream>
#include<string>
#include<map>
using namespace std;
int main()
{
    map<string,string>map;
    string a,b,s;
    char str[15];
    while(gets(str)&&strlen(str))
    {
        a=b="\0";
        int i;
        for(i=0;;i++)
            if(str[i]==' ')
            {
                str[i]='\0';
                break;
            }
        a+=str;
        b+=str+i+1;
        map[b]=a;
    }
    while(cin>>s)
    {
        if(map[s].size())
            cout<<map[s]<<endl;
        else
            cout<<"eh"<<endl;
    }
    return 0;
}
```