

# 2020牛客暑期多校训练营（第七场）

[比赛网址](#)

## 训练结果

- 时间: ' ' 2020-08-01 12:00~17:00 ' '
- rank: "196/1090"
- 完成情况: ' ' 3/3/10 ' '

## 题解

### B.Mask Allocation

#### 题意

有  $n \times m$  个物品，让你给一个划分方案，使得既能分成  $n$  组  $m$  个又能分成  $m$  组  $n$  个

#### 题解

发现最优划分类似于 gcd 的做法，若  $n < m$ ，我们就用  $n$  去补  $m$  补不了的部分就转化为了  $m \% n$  的问题，递归处理。

```
#include<iostream>
#include<cstdio>
#include<algorithm>
#include<cstring>
#define ll long long
using namespace std;
int read()
{
    int k=0,f=1;char c=getchar();
    for(;!isdigit(c);c=getchar()) if(c=='-') f=-1;
    for(;isdigit(c);c=getchar()) k=k*10+c-'0';return k*f;
}
int T,n,m,tot,ans[100055];
void solve(int x,int y)
{
    if(x>y) swap(x,y);
    if(!x) return;
    int res=y/x*x;
    for(int i=1;i<=res;i++)
        ans[++tot]=x;
}
```

```
    solve(y%x,x);
}
bool cmp(int x,int y)
{
    return x>y;
}
int main()
{
    for(T=read();T;T--)
    {
        n=read();m=read();
        tot=0;solve(n,m);
        sort(ans+1,ans+1+tot,cmp);
        printf("%d\n",tot);
        for(int i=1;i<=tot;i++)
            printf("%d ",ans[i]);
        puts("");
    }
    return 0;
}
```

## D.Fake News

### 题意

问平方和是否为完全平方数

### 题解

开始用python写高精度结果T了（人家用cin都T了更何况py写了个表发现就只有1和24满足

## H.Dividing

### 题意

一下数对是合法的

1、 $(1,k)$ 是合法的

2、如果 $(n,k)$ 合法，那么 $(n+k,k)$ 合法

3、如果 $(n,k)$ 合法，那么 $(nk,k)$ 合法

给定范围 $n,k \leq 10^{12}$ ，求范围内所有合法的数对

## 题解

容易发现 $k$ 是不变的，那么只需对每个 $k$ 求出 $n$ 的个数

容易发现所有 $(1+tk, k)$ 都是合法的

除此之外所有合法的必然是 $k$ 的倍数，而且 $k$ 的倍数一定能被凑出来

因此合法的只有 $(tk, k)$ 和 $(1+tk, k)$ 其中 $t$ 为非负整数

只需要整除分块即可

记得特判 $n, k$ 分别为 $1$ 的情况 = =

```
#include<algorithm>
#include<iostream>
#include<cstdlib>
#include<cstring>
#include<cstdio>
#include<vector>
#include<queue>
#include<cmath>
#include<map>
#include<set>
#define LL long long int
#define REP(i,n) for (int i = 1; i <= (n); i++)
#define Redge(u) for (int k = h[u], to; k; k = ed[k].nxt)
#define cls(s,v) memset(s,v,sizeof(s))
#define mp(a,b) make_pair<int,int>(a,b)
#define cp pair<int,int>
using namespace std;
const int maxn = 100005, maxm = 100005, INF = 0x3f3f3f3f, P = 1000000007;
inline LL read(){
    LL out = 0, flag = 1; char c = getchar();
    while (c < 48 || c > 57){if (c == '-') flag = 0; c = getchar();}
    while (c >= 48 && c <= 57){out = (out << 1) + (out << 3) + c - 48; c =
getchar();}
    return flag ? out : -out;
}
LL N, K, ans = 0;
int main(){
    cin >> N >> K;
    if (K == 1){cout << N % P << endl; return 0;}
    if (N == 1){cout << K % P << endl; return 0;}
    ans = (N + N + K - 2) % P;
    for (LL i = 3, nxt; i <= K && i <= N; i = nxt + 1){
        nxt = N / (N / i);
        if (nxt > K) nxt = K;
        ans = (ans + 1ll * (N / i) % P * ((nxt - i + 1) % P) % P) % P;
    }
}
```

```
N--;
for (LL i = 3, nxt; i <= K && i <= N; i = nxt + 1){
    nxt = N / (N / i);
    if (nxt > K) nxt = K;
    ans = (ans + 1ll * (N / i) % P * ((nxt - i + 1) % P) % P) % P;
}
N++;
ans = (ans % P + P) % P;
cout << ans << endl;
return 0;
}
```

## I.Valuable Forests

### 题意

定义一个无根树的权值为所有点的度数的平方的和，求有标号的 $n$ 个点形成的所有森林的权值的和。

### 题解

设 $f[i]$ 表示 $i$ 个带标号的节点有多少种不同的森林，递推式为 $f[i]=\sum_{j=1}^i C_i^j * f[i-j] * j^{j-2}$

设 $w[i]$ 表示 $i$ 个点能形成的所有无根树的权值和,考虑每一个点以及不同的度数对答案的贡献，递推式子为 $w[i]=i * \sum_{d=1}^{i-1} d^2 * C_{i-1}^{d-1} * (n-1) * (n-2-(d-1))$

那么最终答案 $ans[i]$ 就是 $ans[i]=\sum_{j=1}^i C_i^j * (j^{j-2} * ans[i-j] + f[i-j] * w[i])$

```
#include<bits/stdc++.h>
using namespace std;
#define mem(a,b) memset(a,b,sizeof(a))
typedef long long LL;
typedef pair<int,int> PII;
#define X first
#define Y second
inline int read()
{
    int x=0,f=1;char c=getchar();
    while(!isdigit(c)){if(c=='-')f=-1;c=getchar();}
    while(isdigit(c)){x=x*10+c-'0';c=getchar();}
    return x*f;
}
const int maxn=5010;
int T;
LL MOD,c[maxn][maxn],w[maxn],f[maxn],ans[maxn];
LL quickpow(LL a,int N)
{

```

```

    if(N<=0)return 1;
    LL res=1,tmp=a;
    while(N)
    {
        if(N&1)res=(res*tmp)%MOD;
        tmp=(tmp*tmp)%MOD;
        N>>=1;
    }
    return res;
}
int main()
{
    T=read();MOD=read();
    for(int i=0;i<=5000;i++)c[i][0]=1;
    for(int i=1;i<=5000;i++)
        for(int j=1;j<=i;j++)c[i][j]=(c[i-1][j]+c[i-1][j-1])%MOD;
    for(int i=3;i<=5000;i++)
    {
        w[i]=i;
        for(int
d=1;d<=(i-1);d++)w[i]=(w[i]*d*d%MOD*c[i-2][d-1]%MOD*quickpow(i-1,i-1-
d))%MOD;
    }
    printf("here");
    f[0]=1;f[1]=1;f[2]=2;
    for(int i=3;i<=5000;i++)
        for(int j=1;j<=i;j++)
            f[i]=(f[i]+c[i][j]*f[i-j]%MOD*quickpow(j,j-2)%MOD)%MOD;
    ans[1]=0;ans[2]=2;
    for(int i=3;i<=5000;i++)
        for(int j=1;j<=i;j++)
            ans[i]=(ans[i]+c[i][j]*(quickpow(j,j-2)*ans[i-j]%MOD+f[i-
j]*w[j]%MOD)%MOD)%MOD;
    while(T--)printf("%lld\n",ans[read()]);
    return 0;
}

```

## 训练实况

0~1:40 过三题 之后垃圾时间

## 训练总结

wxg: 这场发挥很差，c一直执着于一个错误算法。

hxm:菜是原罪，\$H\$调了很久才调出来特判没判，\$A\$是模拟退火但却不能正确写对

fyh:是真的读不懂题，第一英语差，第二额是本身对指针，对象之类的玩意不熟悉。以后帮队友调题的时

候要注意几点：特殊情况 $n=1$ 等等是否取模！！)

From:

<https://wiki.cvbbacm.com/> - **CVBB ACM Team**

Permanent link:

[https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:die\\_java:front\\_page\\_summertrain8](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:die_java:front_page_summertrain8) 

Last update: **2020/08/20 10:56**