

# Character Encoding

## 一. 考察内容:

数论 容斥原理

## 二. 题目分析:

[题目大意]

给出一个长度为 $m$ 的序列，序列中每个位置可以填入一个不超过 $n$ 的数，求出使得这些数之和为 $k$ 的方案数。

[写题思路]

考虑用容斥原理实现。

如果我们不限制数的大小不能达到 $n$ ，则方案数为 $C(m+k-1, m-1)$ （隔板法，把隔板和物品混在一起，有 $m+k-1$ 个可以放隔板的地方，一共放 $m-1$ 个隔板），再减去有一个位置的数不满足大小要求的情况： $C(n+k-1-n*1, m-1)*(m, 1)$ ，再加上有两个位置不满足大小要求的情况： $C(n+k-1-n*2, m-1)*(m, 2) \dots$ ，用容斥原理算出最后结果即可。

## 三. 代码实现:

```
#define _CRT_SECURE_NO_DEPRECATE
/*****
*创建时间: 2018 08 24
*文件类型: 源代码文件
*题目来源: HDU
*当前状态: 已通过
*备忘录: 容斥原理 数论
*作者: HtBest
*****/
#include <stdio.h>
#include <iostream>
#include <string>
#include <stdlib.h>
#include <string.h>
#include <algorithm>
#include <queue>
// #include <sys/wait.h>
// #include <sys/types.h>
// #include <unistd.h>
using namespace std;
#define MOD 998244353
int n, m, k, fac[200010], faci[200010];
/* Variable explain:

*/
void read()
{
    scanf("%d%d%d", &n, &m, &k);
    return;
}
int add(int a, int b)
{
    return ((long long)a+b+MOD)%MOD;
}
int mul(int a, int b)
{
    return (long long)a*b%MOD;
}
int exgcd(int a, int b, int &x, int &y)
{
    if(!b)
    {
```

```
        x=1,y=0;
        return a;
    }
    int tmp=exgcd(b,a%b,y,x);
    y-=a/b*x;
    return tmp;
}
int inverse(int a)
{
    int x,y;
    exgcd(a,MOD,x,y);
    return add(x,MOD);
}
int C(int a,int b)
{
    if(a<b) return 0;
    return mul(fac[a],mul(faci[a-b],faci[b]));
}
void solve()
{
    int t=min(k/n,m),ans=0;
    for(int i=0;i<=t;++i)
    {
        ans=add(ans,((i&1)?-1:1)*mul(C(m+k-1-n*i,m-1),C(m,i)));
    }
    printf("%d\n",ans);
}
void init()
{
    fac[0]=1;
    for(int i=1;i<=200001;++i) fac[i]=mul(fac[i-1],i);
    for(int i=0;i<=200001;++i) faci[i]=inverse(fac[i]);
}
int main()
{
    // freopen(".in","r",stdin);
    // freopen(".out","w",stdout);
    int T;
    scanf("%d\n",&T);
    init();
    while(T--)
    {
        read();
        solve();
    }

    return 0;
}
```

[<题目跳转>](#) [<查看代码>](#)