

Pluses everywhere

一. 考察内容:

组合数取模 逆元

二. 题目分析:

[题目大意]

给出一个数字串，在数字串中添加 k 个“+”号，使其凑成合法的算式，并求出算式结果，输出所有添加“+”号的方案的算式结果之和。

[写题思路]

对于第 i （从左到右计算，第一个数字是0位）位上的数字：

如果他的后面至少有一个加号，那么这个加号可以填在 i 向后数 $0 \sim n-i-2$ 的任意一位后面，设加号填在 i 向后数 j 位，这时第 i 位数代表 $a_i * 10^j$ ，还需要在其他 $n-j-2$ 个位置放上 $k-1$ 个加号，方案数为 $\binom{n-j-2}{k-1}$ ，总贡献为 $\sum a_i * 10^j * \binom{n-j-2}{k-1}$ 。

如果他后面没有加号，那么该数字代表 $a_i * 10^{n-i-1}$ ，其他的 k 个加号需要填在这个数前面，填加号的方案数为 $\binom{i}{k}$ 。

对于每一个数，求出这两种情况的方案数之和，化简公式：

$$\sum_{i=0}^{n-2} \sum_{j=0}^{n-i-2} a_i 10^j \binom{n-j-2}{k-1} = \sum_{j=0}^{n-2} \sum_{i=0}^{n-j-2} a_i 10^j \binom{n-j-2}{k-1} = \sum_{j=0}^{n-2} 10^j \binom{n-j-2}{k-1} \sum_{i=0}^{n-j-2} a_i$$

三. 代码实现:

```
#define _CRT_SECURE_NO_DEPRECATED
/*****
*创建时间：2018 08
*文件类型：源代码文件
*题目来源：CodeForces
*当前状态：已通过
*备忘录：组合数 逆元 数论
*作者：HtBest
*****/
#include <stdio.h>
#include <iostream>
#include <string>
#include <stdlib.h>
#include <string.h>
#include <algorithm>
#include <queue>
using namespace std;
#define MAXN 100010
#define MOD 1000000007
int n, k, a[MAXN], fac[MAXN], sum[MAXN];
/* Variable explain:

*/
void read()
{
    // freopen(".in", "r", stdin);
    // freopen(".out", "w", stdout);
    char s[MAXN];
```

```

    scanf("%d%d%s",&n,&k,s);
    for(int i=0;i<n;++i)a[i]=s[i]-'0';
    return;
}
int add(int a,int b)
{
    return (a+b)%MOD;
}
int mul(int a,int b)
{
    return (long long)a*b%MOD;
}
int pow(int a,int n)
{
    int ans=1;
    while(n)
    {
        if(n&1)ans=mul(ans,a);
        a=mul(a,a);
        n>>=1;
    }
    return ans;
}
int exgcd(int a,int b,int &x,int &y)
{
    if(b==0)
    {
        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(inverse(fac[a-b]),inverse(fac[b])));
}
int solve()
{
    int ans=0;
    sum[0]=a[0];
    for(int i=1;i<n;++i)sum[i]=add(a[i],sum[i-1]);
    for(int i=0;i<=n-2;++i)
    {
        ans=add(ans,mul(sum[n-i-2],mul(pow(10,i),C(n-i-2,k-1))));
    }
    for(int i=0;i<n;++i)ans=add(ans,mul(a[i],mul(pow(10,n-i-1),C(i,k))));
    return ans;
}
void init()
{
    fac[0]=1;
    for(int i=1;i<=100000;++i)fac[i]=mul(fac[i-1],i);
}
int main()
{
    init();

```

```
    read();  
    printf("%d\n", solve());  
    return 0;  
}
```

[<题目跳转>](#) [<查看代码>](#)