**1.数学（math.pas/c/cpp）**

**来源：bzoj4870 SHOI2017 组合数问题 加强版**

1-3：O（N^2）预处理+O（N）询问。

4-7：用求逆元的方法将预处理优化到O（NlogN）或O（N）

8-10 发现答案=sigma(C(n,i))=2^n

O（logn）快速次幂

11-13发现不论r是多少答案都是sigma（C(n-1,i)）=2^(n-1)

O（logn）快速次幂

14-19 显然是可以dp的辣。

设f[i][j]表示sigma（C（i,pk+j））,p为正整数

发现f[i][j]=f[i-1][j]+f[i-1][(j-1+k)%k]

显然这一坨东西可以用矩阵乘法优化掉。

时间复杂度O（TK^3logN）（原题）

20 看着k^3不优美。

我们发现原矩阵是循环矩阵

即满足f[0][j]=f[i][(j+i)%k]

于是我们枚举左边取几个，右边取几个就可以了。

时间复杂度O（TK^2logN）

**2.线段树（seg.pas/c/cpp）**

**来源：bzoj2908又是nand（改）**

1-4： 直接O（NQ）的超级大暴力水过去。

5-8： 根据~~标题~~我们推断可以用线段树做。

但是nand不满足交换律。

我们对于每个节点维护每一位为1/0时的返回值

每一次我们大力合并。

询问我们按位考虑。

时间复杂度O（NlogN\*32）

~~当然在常数比较小的情况下是可以通过的~~

9-10：出题人都和你说了小心常数问题啦。

所以上面的算法是要被卡常数的。

因为题目告诉你是线段树了

所以还是在32上下功夫吧

我们将32个bool强制压成一个unsigned int

然后我们就可以O（1）合并了

时间复杂度O（NlogN）

**3.字符串（string.pas/c/cpp）**

**来源：某天突发奇想**

1-5直接O（Q\*C（n,k）\*N\*|S|）捻过去

6-10 预处理一下O（Q\*C（n,k）\*N^2+N\*|S|）跑过去

11-14建出字母树。

显然每个非根节点对答案的贡献是C(sum,2)

我们预处理出每一个非根节点数量少1时答案减少的数量

然后删掉一个字符串，答案就会减少从跟到末节点的的所有减少的数量之和

这个显然可以前缀和实现

O（N+|S|）

15-18 每次询问我们考虑每个节点对答案的贡献

我们枚举经过这个节点的字符串到底减少了几个，假设为x

我们同时假设原来有Y个字符串经过这个点

那么他的贡献就是C（sz,x）\*C(n-sz,k-x)\*(k-x)\*(k-x-1)/2

直接求和即可

O（|S|\*Q\*K）

19-20 我们直接算出sz为i的节点个数

然后我们是要枚举i就可以了

时间复杂度O（N\*Q\*K）