

## D. KEVIN'S PROBLEM

### PROBLEM DESCRIPTION

題目給  $N, K, p$ ，問  $N$  的全排列中，有幾種排列符合：從第  $K+1$  項開始看，第一個比前面數字大的數字是  $p$ ，或是從  $K+1$  項開始遞增的，且  $p$  是最後一個數字。

### SOLUTION TECHNIQUES

數學、排列組合

### SOLUTION SKETCHES

有兩種 Case 分開討論。

第一種 Case 是  $p$  前面有比較大的數字存在，對於這個 Case 我們可以先枚舉比  $p$  大的數字為何，接著再枚舉  $p$  在哪個位置上：

$$\sum_{i=p+1}^n \sum_{l=0}^{n-k} \binom{i-2}{l} (n-l-2)!$$

第二種 Case 是  $p$  為末尾且前面序列為遞增，這個 Case 比較簡單： $p-1$  個數字中選  $n-k$  個按照順序排前面，剩下  $k-1$  個數字隨便排。

$$\binom{p-1}{n-k} (k-1)!$$

### TIME COMPLEXITY

每筆測資  $O(N^2)$

## SOLUTION PROGRAM FOR REFERENCE

```
#include <stdio>
#include <string>
const int MOD = 1e9+7;
int com[502][502];
int fact[502]={1};
int C(int n, int k)
{
    if(n<0||k<0||n<k) return 0;
    if(k==0||n==k) return 1;
    if(com[n][k]!=-1) return com[n][k];
    return com[n][k]=(C(n-1,k)+C(n-1,k-1))%MOD;
}
int main()
{
    memset(com,-1,sizeof(com));
    for(int i=1;i<502;i++)
        fact[i]=1LL*fact[i-1]*i%MOD;
    int T,n,k,p;
    scanf("%d",&T);
    for(int cases=1;cases<=T;cases++)
    {
        scanf("%d%d%d",&n,&k,&p);
        int ans=0;
        for(int i=p+1;i<=n;i++)
        {
            for(int j=k,l=0;j<n;j++,l++)
            {
```

```

        int t=C(i-2,l);
        //printf("t %d\n",t);
        t=1LL*t*fact[n-2-l]%MOD;
        ans=(ans+t)%MOD;
    }
}
int t=1LL*C(p-1,n-k)*fact[k-1]%MOD;
ans=(ans+t)%MOD;
printf("Case #%d: %d\n",cases,ans);
}
}

```