

用途

统计 [l,r] 内符合要求的数字，一般是数位上带有xx的数字。

思路

(一般性的思路) 设 $dp[i][j]$ 为不超过 i 位的数字数位上含有数字 j 的个数 $(0 \leq j \leq 9)$ ，则 $dp[i][j] = 10 \times dp[i-1][j] + 10^{i-1}$ 。首先在第 i 位加上前导的数字 0 到 9 即乘十，暂不考虑前导数字中对 j 的统计；而前导数字为 j 的情况下，不考虑前 $i-1$ 位数字中对 j 的统计(上一步已经统计过)，则是前 $i-1$ 位数字数目。

一般的题目中仍需要更多动态规划的构思。

例题

洛谷 p2602 数字计数

给定两个正整数 a 和 b 求在 $[a,b]$ 中的所有整数中，每个数码(digit)各出现了多少次。对于一个数 x ，计算 $[0,x]$ 中每个数码出现的次数：对于最高位 i ，首先每个数码都出现了 $i \times dp[i-1]$ 次，其次从 0 到 $i-1$ 的每个数码都出现了 10^{i-1} 次；对于第 i 位， i 这个数码出现了去掉这一位之后余下的数字加一的次数；而去掉这一位之后余下的数字中的数码各自出现的次数需要 $\times 1$ 即按照这种方法计算下一位。

```
#include<cstdio>
using namespace std;
long long a,b,dp[20],po[20]={1},all_=0,cnt[20];
void count(long long x,int m)
{
    if(!x) cnt[0]+=m;
    long long num=0;
    for(int i=0,op=x%10;x;num+=op*po[i++],x/=10,op=x%10)
    {
        for(int j=0;j<op;j++)
            cnt[j]+=po[i]*m;
        all_+=op*dp[i]*m;
        cnt[op]+=(num+1)*m;
        if(i) cnt[0]-=po[i]*m;
    }
}
int main()
{
    scanf("%lld%lld",&a,&b);
    for(int i=1;i<=15;i++)
    {
```

```
    po[i]=po[i-1]*10;  
    dp[i]=dp[i-1]*10+po[i-1];  
}  
count(b,1);  
count(a-1,-1);  
for(int i=0;i<=9;i++)  
    printf("%lld ",all_cnt[i]);  
return 0;  
}
```

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

Permanent link: https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:no_morning_training:shaco:%E7%9F%A5%E8%AF%86%E7%82%B9:%E5%8A%A8%E6%80%81%E8%A7%84%E5%88%92:%E6%95%B0%E4%BD%8Ddp&rev=1594895848

Last update: 2020/07/16 18:37