

递推

$$\begin{aligned}
f_{k+1}(n+1) &= f_k(n) + (n+1)^{k+1} \\
&= 1 + \sum_{i=1}^n (i+1)^{k+1} \\
&= 1 + \sum_{i=1}^n \sum_{j=0}^{k+1} \binom{k+1}{j} i^j \\
&= 1 + \sum_{j=0}^{k+1} \binom{k+1}{j} \sum_{i=1}^n i^j \\
&= 1 + f_{k+1}(n) + (k+1)f_k(n) + \sum_{i=0}^{k+1} \binom{k-1}{i} f_i(n)
\end{aligned}$$

拉格朗日插值法

对于 $\sum_{i=1}^n i^k = f[n][k]$ 具体表达是一个 $k+1$ 次多项式，我们求解 $f[0 \sim k+1][k]$ 然后进行插值即可

对于具体计算 $f[n] = \sum_{i=0}^{k+1} f[i] \frac{\prod_{j=0, j \neq i}^{k+1} (n-j)}{\prod_{j=0, j \neq i}^{k+1} (i-j)}$

$\prod_{j=1, j \neq i}^{k+1} (n-j) = \prod_{j=1}^{i-1} (n-j) \prod_{j=i+1}^{k+1} (n-j)$ 是前缀积 \times 后缀积

$$\prod_{j=0, j \neq i}^{k+1} (i-j) = i!(k+1-i)!(-1)^{k+1-i}$$

于是，可以 $O(k)$ 预处理 $f[i]$ 以及阶乘，前后缀然后 $O(k)$ 计算

这种方法对不同 $f[n]$ 计算速度一样，而针对多次询问则需要利用生成函数计算伯努利数

伯努利数以及生成函数

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

Permanent link: <https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:alchemist:hardict:powersum&rev=1588990015>

Last update: 2020/05/09 10:06