

递推

令 $f_k(n) = \sum_{i=1}^n i^k$

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

得到了: $(k+1)f_k(n) = (n+1)^{k+1} - 1 - \sum_{i=0}^{k-1} \binom{k+1}{i} f_i(n)$

可以写成: $f_{k-1}(n) = \frac{(n+1)^k - 1}{k} - \frac{1}{k} \sum_{i=0}^{k-2} \binom{k}{i} f_i(n)$

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

$B_0 = 1, B_1 = -\frac{1}{2}, B_2 = \frac{1}{6}, B_3 = 0, B_4 = -\frac{1}{30}, B_5 = 0, B_6 = \frac{1}{42}, B_7 = 0, B_8 = -\frac{1}{30}$

可以利用: $B_0 = 1, \sum_{i=0}^n B_i \binom{n+1}{i} = 0 \Rightarrow B_n = -\frac{1}{n+1} \sum_{i=0}^{n-1} B_i \binom{n+1}{i}$ 计算

$$\begin{aligned} \sum_{i=0}^{n-1} B_i \binom{n}{i} &= 0 \\ \sum_{i=0}^n B_i \binom{n}{i} &= B_n \quad (n > 1) \\ \sum_{i=0}^n \frac{1}{i!(n-i)!} B_i &= \frac{B_n}{n!} \quad (n > 1) \end{aligned}$$

考虑 $C(x) = \sum_{i=0}^{\infty} \frac{B_i}{i!} x^i$
有 $e^x C(x) = C(x) + x \text{quad}$ (加 x 是因为 $n > 1$ 导致第一项缺失, 而 $n = 0$ 时上述等式也是成立的)
得到关于 $\frac{B_n}{n!}$ 的生成函数 $\frac{x}{e^x - 1}$

故可以利用递推 $O(n^2)$ 或利用 NTT $O(n \log n)$ 预处理

伯努利数与自然数幂和关系为

$$f_{k}(n-1) = \sum_{i=0}^{n-1} i^k = \frac{k+1}{k+1} \sum_{i=0}^k \binom{k+1}{i} B_i n^{k+1-i}$$

From:

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

Permanent link:

<https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:alchemist:hardict:powersum&rev=1588992091> 

Last update: **2020/05/09 10:41**