

第 23 届全国青少年信息学奥林匹克联赛

CCF NOIP 2017 提高组

解题报告

Kay

题目名称	数列求和	路径统计	小店购物
题目类型	传统型	传统型	传统型
目录	sum	road	shopping
可执行文件名	sum	road	shopping
输入文件名	sum.in	road.in	shopping.in
输出文件名	sum.out	road.out	shopping.out
每个测试点时限	1.0 秒	1.0 秒	2.0 秒
内存限制	256MB	256MB	256MB
测试点/子任务数目	10	20	10
每个测试点分值	10	5	10

提交源程序文件名

对于 C++ 语言	sum.cpp	road.cpp	shopping.cpp
对于 C 语言	sum.c	road.c	shopping.c
对于 Pascal 语言	sum.pas	road.pas	shopping.pas

注意：最终测试时，不开启 O2 优化，栈空间大小与内存限制相同。

数列求和(sum)

【问题大意】

给你一个长度为 n 的数列 a_1, a_2, \dots, a_n , 求 $\sum_{i=1}^n \sum_{j=i}^n \prod_{k=i}^j a_k \bmod p$ 。

【30 分算法】 ($n \leq 100$)

按题目意思枚举 i, j, k , 直接求和即可, 复杂度 $O(n^3)$ 。

【60 分算法】 ($n \leq 1000$)

已知 $\langle i, j \rangle$ 的答案, 可以 $O(1)$ 求出 $\langle i, j+1 \rangle$ 的答案, 所以可以边枚举边求积, 复杂度 $O(n^2)$ 。

【80 分算法】 ($n \leq 100000, p \leq 10^9$)

考虑所有右端点为 x 的答案:

$$a_x, a_{x-1}a_x, a_{x-2}a_{x-1}a_x, \dots, a_1a_2\dots a_x。$$

不难发现, 答案为所有右端点为 $x-1$ 的答案加 1, 再乘 a_x 。这样我们就可以从左到右依次递推求出以每个点为右端点的答案, 最后累加即可, 复杂度 $O(n)$ 。

【100 分算法】 ($n \leq 100000, p \leq 10^{18}$)

相比 80% 的数据, 只是模数的范围扩大到了 10^{18} 级别, 这时就不能直接乘起来取模, 但这也不难处理, 使用快速乘算法即可, 时间复杂度 $O(n \log A)$ 或 $O(n)$, 空间复杂度 $O(1)$ 。

路径统计(road)

【问题大意】略

【30 分算法】 ($n \leq 2000$)

把图建出来，直接跑 Floyd 或 SPFA，或者直接 dfs，因为每条路径是唯一的，复杂度 $O(n^3)$ 或 $O(n^2)$ ，期望得分 15-30 分。

【70 分算法】

首先，不难发现这张图就是一个基环外向树组成的森林，这样，剩下的 70 分就不难了。

【额外 20 分算法】 ($len_i = 0$)

我们只需求出可达点对个数即可。对于每个基环外向树，我们把环上的每个点看做每棵子树的根节点，这样，每个点可以到达的点即为这个点到根的路径上的点和环上的点，根据深度信息即可算出。

【额外 20 分算法】 (t_i 互不相同)

说明 t_i 构成了一个排列，这样这个图就由若干个环组成，那么每条边对答案的贡献就很好求了，再算出不可达点对个数即可。

【满分算法】 ($n \leq 500000$)

我们参考 11-14 测试点的做法，考虑每条边对答案的贡献。对于树边，则下方子树与上方路径上的点和环上点的路径经过这条边；对于环上的边，求解方式与 11-14 测试点相同。最后算出不可达点对个数。时间复杂度 $O(n)$ 。

小店购物(shopping)

【问题大意】略

【0 分算法】

一共只有 k 元钱, 所以买的商品个数不会超过 k , 复杂度 $O(nmk)$ 。

应该可以通过第一个测试点。

【20 分算法】

不妨将寻找商品的复杂度降下来, 将商品按价格排序, 预处理价值的前缀最大值, 查找时直接二分即可, 修改是暴力更新, 复杂度是 $O(mn\log n + mk\log n)$ 。

【额外 20 分算法】

不存在修改, 这样复杂度就是询问的复杂度 $O(mk\log n)$ 。考虑初始时拥有 k 元钱, 第一次买的商品价格为 p , 若 $k-p \geq p$, 则第二次买的商品依然是该商品, 所以该商品一共会买 $\text{floor}(k/p)$ 件, 剩下 $k \bmod p$ 元。又存在一个结论: 若 $k \geq p$, 则 $k \bmod p < k/2$, 所以一共只会买 \log 种商品, 复杂度也就是 $O(n\log n + m\log n \log k)$ 。

【满分算法】

加入修改的话, 就用数据结构维护所有商品, 需要支持单点修改、区间最值, 用平衡树维护即可。本题并没有强制在线, 可以离散化后用常数较小的线段树, 复杂度 $O(m\log n + m\log n \log k)$ 。