Za 选串串 (B)

Problem A: [NOI 2015] 品酒大会

给定一个字符串 $S_{1...n}$ 以及一个权值序列 $a_{1...n}$。

对于 $r = 0 \dots n-1$，分别求出：

- $\sum_{1 \le i,j \le n} [S_{i...i+r-1} = S_{j...j+r-1}]$；
- $\max a_i a_j \ \ (S_{i...i+r-1} = S_{j...j+r-1})$。

即有多少对后缀的 LCP 长度至少为 r，以及满足条件的位置对的权值积的最大值。

$n \le 3 \times 10^5$，$|a_i| \le 10^9$，$\Sigma = \{\mathtt{a} \dots \mathtt{z}\}$。

Source: https://uoj.ac/problem/131

考虑 SA 的 ht 序列。

考虑如何求一个给定的 r 的答案。

按 $\mathrm{ht} < r$ 的位置将 SA 分段，则第一问的答案即 $\sum \binom{\mathrm{len}}{2}$。

第二问的答案可通过记录每段内的最大值、次大值、最小值、次小值求出。

不难发现，若从大到小枚举 r，则分段只会合并而不会分裂。

合并时维护两问答案即可。时间复杂度 $O(n \log n)$。

Problem B: [SDOI 2008] Sandy 的卡片

给定 N 个整数串，第 i 个长度为 M_i，求其相似意义下最长公共子串。

两个整数串 s, t 相似，当且仅当 $|s| = |t|$ 且 $\forall i, j$ 都有 $s_i - t_i = s_j - t_j$。

$40 \le N \le 10^3$，$2 \le M_i \le 101$，值域为 $[0, 1864] \cap \mathbb{Z}$。

Source: https://www.luogu.com.cn/problem/P2463

Solution A

将差分数组用不同的特殊字符连接起来建 SA。

则我们要找到 SA 中的一个区间，满足区间中包含来自全部 N 个串的位置，并最小化区间的 LCP 长度。

枚举区间右端点，对每个串维护其最后出现位置即可。

时间复杂度 $O(\sum M_i \log(\sum M_i))$。

Solution B

二分 + 哈希即可。

时间复杂度 $O(\sum M_i \log(\sum M_i))$。

Problem C: [JZOI-1] 拜神

给定长为 n 的字符串 s。

q 次询问，每次给定 l, r，求最长的字符串 t 的长度，满足：

- t 在 $s_{l...r}$ 中至少出现了两次。

$1 \le n \le 5 \times 10^4$，$1 \le q \le 10^5$。

Source: https://www.luogu.com.cn/problem/P7361

Solution A

考虑 SA。

从大到小枚举 $|t| = k$，按 $\text{ht} < k$ 的位置将 SA 分段。

则 $s_{l...r}$ 的答案至少为 k 当且仅当 $l \ldots r-k+1$ 所属的段互不相同。

判断一个区间内元素互不相同的经典方法之一是记录每个元素的前驱。

现在要在合并两段的同时维护前驱。可以启发式合并。

询问的时候二分答案即可。

由于要保留每个版本的信息，所以需要用可持久化数据结构记录前驱。

时间复杂度 $O((n+q)\log^2 n)$，空间复杂度 $O(n\log^2 n)$。

Solution B

考虑 SAM。

考虑 $\text{ans}(l,r)=\max_{l\le i<j\le r}\min\{i-l+1,\text{LCS}(s_{1\dots i},s_{1\dots j})\}$。

考虑 endpos 集合合并的过程不难发现，可能有贡献的 (i,j) 对数为 $O(n\log n)$。

记录所有可能有贡献的 (i,j,LCS) 三元组。

对右端点扫描线，就可以只考虑 $j\le r$ 的情况了。

此时，一个三元组 (i,j,LCS) 会对 $l=1\dots i$ 产生 LCS 的贡献，会对 $l=i+1\dots n$ 产生 $i-l+1$ 的贡献。分别线段树维护即可。

时间复杂度 $O(n\log^2 n+q\log n)$。

Problem D: [CTSC 2012] 熟悉的文章

给定 M 个 01 串构成的字典。有一个参数 L。

称一个 01 串 A「出现过」，当且仅当 $|A|\ge L$ 且 A 是字典中某个串的子串。

称一个 01 串 A 是「熟悉的」，当且仅当它可被分解为 $A=a_1\dots a_k$，其中「出现过」的 a_i 的长度和至少为 $0.9|A|$。

给定 N 个 01 串，分别求出最大的 L，使得它们是「熟悉的」。

输入的 01 串总长不超过 1.1×10^6。

Source: https://www.luogu.com.cn/problem/P4022

二分答案 L。

记询问串为 S。设 f_i 表示将 $S_{1\dots i}$ 分解后的最大的「出现过」的长度和。

记 g_i 表示 $S_{1\dots i}$ 的最长的「出现过」的后缀长度，则有：

$$f_i = \max\{f_{i-1}, \max_{j=i-g_i}^{i-L} f_j + (i-j)\}$$

$g_{1\dots|S|}$ 则可以通过 SA 或 SAM 或二分 + 哈希求出。

- SA：将字典与询问串 reverse 后用特殊字符连接建 SA，考察 SA 中上一个字典内的串和下一个字典内的串与当前串的 LCP 即可。
- SAM：将字典用特殊字符连接建 SAM，然后在上面跑询问串，考察当前到达的节点的 len 即可。
- GSAM：类似 SAM，不过直接建即可。

设所有串长之和为 T，线段树维护 f_i 进行转移即可做到 $O(T\log^2 T)$。

注意到转移方程中 $i-g_i \dots i-L$ 这个区间的左右端点均单调不降。

于是可以单调队列，时间复杂度为 $O(T\log T)$。

Problem E: [LNOI 2022] 串

给定字符串 S。

你需要找到一个字符串序列 $T_{0\dots l}$，满足：

- T_0 是 S 的子串（特别地，可以为空）；
- $\forall 1\le i\le l, |T_i|-|T_{i-1}|=1$，即串长递增；
- $\forall 1\le i\le l$，存在 S 的一个长度为 $|T_i|+1$ 的子串 s，使得 s 的长为 $|T_{i-1}|$ 的前缀为 T_{i-1}，长为 $|T_i|$ 的后缀为 T_i。

换句话说，若 $T_i=S_{l\dots r}$，则 T_{i+1} 可以为 $S_{l+1,r+2}$。

最大化 l，求这个最大值。多测，T 组数据。

$T\ge 1$，$1\le |S|\le 5\times 10^5$，$1\le \sum|S|\le 1.5\times 10^6$，$\Sigma=\{\texttt{a}\dots\texttt{z}\}$。

Source: https://loj.ac/p/3740

比较平凡的构造是 $\epsilon\to(2,2)\to(3,4)\to\dots$，这样可以做到 $l=\lfloor\frac{n}{2}\rfloor$。

若 $l>\lfloor\frac{n}{2}\rfloor$，则必定存在某个 T_i 在 S 中出现了至少两次。

假设最后一个出现了两次的 T_i 是 $T_i = S_{l_0\dots r_0} = S_{l_1\dots r_1}$，则按如下方式构造，必定可以做到 $l = (r_0 - l_0 + 1) + \lfloor \frac{n-r_0}{2} \rfloor$：

- 由于 T_i 是最后一个出现了两次的，所以 T_i 之后必定朴素地构造，即 $(l, r) \to (l+1, r+2)$，这样可构造 $\lfloor \frac{n-r_0}{2} \rfloor$ 个；
- T_i 之前可以从 l_1, r_1 一点点左移，当 $l = l_0$ 时平移到 l_1 处，即可构造 $r_0 - l_0 + 1$ 个。

所以对所有出现了至少两次的子串 (l, r)，求出 $\max\{r - l + 1 + \lfloor \frac{n-r}{2} \rfloor\}$ 即可。

SA 和 SAM 均可轻松计算出该最大值。

Problem F: [NOI 2023] 字符串

给定长为 n 的字符串 s。

q 次询问，每次给定 i, r，求 $1 \dots r$ 中 l 的数目，使得：

$$s_{i\dots i+l-1} < \text{reverse}(s_{i+l\dots i+2l-1})$$

多测，t 组数据。

$1 \le t \le 5$，$1 \le n, q \le 10^5$，$1 \le i + 2r - 1 \le n$，$\Sigma = \{\texttt{a} \dots \texttt{z}\}$。

Source: https://uoj.ac/problem/819

构造字符串 $t = s + \text{chr}(-\inf + 1) + \text{reverse}(s) + \text{chr}(-\inf)$。

则 $s_{i\dots i+l-1} < \text{reverse}(s_{i+l\dots i+2l-1})$ 在多数情况下等价于 $t_{i\dots|t|} < t_{|t|-(i+2l-1)\dots|t|}$。

求出满足后者的 l 的数目可以比较容易地利用 SA 解决。

而二者不相等仅当 $s_{i\dots i+2l-1}$ 是回文串。考虑每个回文串所属的极长偶回文串。

考虑以 c 和 $c+1$ 为中心的一个极长偶回文串 $S_{l\dots r}$。

- 若 $l = 1$ 且 $r = n$，则对答案没有影响；
- 否则，若 $l = 1$，则对答案没有影响；
- 否则，若 $r = n$，则对答案的影响为 -1；
- 否则，若 $s_{l-1} > s_{r+1}$，则对答案的影响为 -1；
- 否则，若 $s_{l-1} < s_{r+1}$，则对答案没有影响。

每个极长偶回文串对 (c,l,r) 对答案的影响可以刻画为二维平面的区间加。可以简单地用二位数点维护。

时间复杂度 $O(\sum(n+q)\log n)$。

Problem G: [CF 700 (Div. 1)] Cool Slogans

给定长为 n 的字符串 w。

你需要找到一个字符串序列 $t_{1...l}$，满足：

- t_i 为 w 的子串；
- t_i 在 t_{i+1} 中至少出现了两次（可部分重叠）。

最大化 l，求这个最大值。

$1 \leq n \leq 2 \times 10^5$，$\Sigma = \{\texttt{a} \dots \texttt{z}\}$。

Source: https://codeforces.com/problemset/problem/700/E

考虑必定存在一个最优的 $t_{1...l}$，满足 t_i 是 t_{i+1} 的 border。

考虑必定存在一个最优的 $t_{1...l}$，满足每个 t_i 都是 SAM 某个节点表示的最长串。

而一个节点（的最长串）能否接在另一个节点（的最长串）前面，可预处理线段树合并后 $O(\log n)$ 求出。于是就可以在 SAM 的 prefix tree 上贪心了。

设 f_x 表示只考虑从根到 x 的链上的点，所能找到的最长 $t_{1...l}$ 的长度。设 g_x 表示在长度最大化的前提下，t_l 对应的最浅点。则有：

$$f_x = f_{\text{link}(x)} + [g_{\text{link}(x)} \to x]$$
$$g_x = \begin{cases} x & g_{\text{link}(x)} \to x \\ g_{\text{link}(x)} & \text{elsewise} \end{cases}$$

其中 $u \to v$ 表示 u 的最长串能否接在 v 的最长串前面。

Problem H: [九省联考 2018] 制胡窜

给定长为 n 的字符串 s。

q 次询问，每次给定 l, r，求 (i,j) 的数目，满足：

- $1 \le i < j \le n$，$j - i \ge 2$；
- $s_{l...r}$ 在 $s_{1...i}, s_{i+1...j-1}, s_{j...n}$ 至少一者中至少出现一次。

$1 \le n \le 10^5$，$1 \le q \le 3 \times 10^5$，$1 \le l \le r \le n$，$\Sigma = \{\texttt{0}\dots\texttt{9}\}$。

Source: https://loj.ac/p/2479

考虑条件即 $s_{l...r}$ 的至少一次出现没有被切开，记 $\text{len} = r - l + 1$。

假设出现位置分别为 $(l_1, r_1) \dots (l_k, r_k)$。讨论以下若干情况：

- 若有三次互不相交的出现，则答案为 $\binom{n-1}{2}$。
- 否则，若 $[l_1, r_1] \cap [l_k, r_k] \ne \varnothing$：

  则未被切开的段必定为一个区间，故答案为：

$$k\binom{n-\text{len}}{2} - \sum_{i=2}^{k}\binom{n-(r_i - r_{i-1} + \text{len})}{2}$$

- 否则：

  令 (l_a, r_a) 为满足 $r_a < l_k$ 的最后一次出现；

  令 (l_b, r_b) 为满足 $l_b > r_1$ 的第一次出现；

  则一种分割方案 (i, j) 不合法仅当 $l_a \le i < r_1$ 且 $l_k < j \le r_b$。

  这种情况下，未被切开的段必为一个区间，故不合法方案数为：

$$\begin{aligned}
&\binom{(r_1 - l_a) + (r_b - l_k)}{2} \\
&-\left(\sum_{i=a}^{b}\binom{(l_i - l_a) + (r_b - r_i)}{2}\right) \\
&+\left(\sum_{i=a+1}^{b}\binom{(l_{i-1} - l_a) + (r_b - r_i)}{2}\right)
\end{aligned}$$

答案在已知 $r_{1...k}$ 即 endpos 集合时总是关于 len 的二次函数。

SAM + 线段树合并维护即可。