FOI2022算法夏令营提高班 DAY8

T1题解

对于前 10% 的数据,暴力即可

容易发现答案是最上最下最左最右四条线构成的矩形的周长

所以如果 $k \geq 4$,答案都相等,可以直接算,可以得到 60 分

考虑对于 $k = 3$的情况,根据抽屉原理,必然有一点是最上/最下/最左/最右,我们分 4 种情况,仅需枚举两个点,双指针排序优化一下即可做到 $O(n \log n)$

T2题解

考虑直接枚举 lcs,设 s 共有字母连续段 x 段,则一共有 x 种 lcs

考虑每种 lcs 要再添加一个字符,一共有 $(n*m)$种添加方式,又注意到对于每个连续段设长为 len,若该连续段字母为 c,那么添加 c 这一字符,就会重复算了 len 次,那么答案要减掉 lcs 的连续段长度之和,即 $n-1$,又因为有一种情况添加后恰和 s 相同,所以还要减 1,所以每个 lcs 可以有 $n*(m-1)$ 种添加方式

接下来我们仅需考虑什么情况下 A 与 s 会有两个不同的 lcs 长度为 $n-1$,用 $x*n*(m-1)$ 减掉这些情况

```
eg:
abab
baba
```

它们的 lcs: aba,bab

我们可以发现:若 A 和 B 的 lcs 有两个,那么它们必然存在一段类似 $abababa$ 或者 $ababab$ 的以2为循环节的子串(也有可能少了最后一个字母)

它们的构成方式即把 A 中的 "$abababab$"部分的最后一个字母挪到第一个字母前,得到 B

所以我们仅需再减掉所有类似的子串个数即可

T3题解

对于30%的数据

考虑直接枚举 i,j 然后求 gcd 再求 ϕ 就完事。

时间复杂度 $O(n^2\sqrt{n})$

对于 50%的数据

考虑预处理 1 到 n 的 ϕ ，然后枚举 i,j 求 gcd 再求 ϕ 。

时间复杂度 $O(n^2\log n)$ 。（其实近似于 $O(n^2)$ 但是辗转相除毕竟复杂度算是 $\log$ 的)

对于 100% 的数据

又到了喜闻乐见的推式子环节。

$\sum_{i=1}^{n}\sum_{j=1}^{n}\phi(gcd(i,j))$

$=\sum_{d=1}^{n}\phi(d)\sum_{i=1}^{n/d}\sum_{j=1}^{n/d}[gcd(i,j)=1]$

$=\sum_{d=1}^{n}\phi(d)*(2*\sum_{i=1}^{n/d}\phi(i)-1)$

这里是使用欧拉函数的意义来推的（就是题面里说的那个互质的个数）。

这样就可以线性预处理 ϕ 然后对于后面那坨整除分块了。

时间复杂度 $O(n+T\sqrt{n})$