



PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE
ESCUELA DE INGENIERIA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACION

Diseño y Análisis de Algoritmos - IIC2283 - 2^{do} semestre - 2023

Tarea 1

Publicación : Miércoles 13 de septiembre
Github Classroom : <https://classroom.github.com/a/vkG1x9XE>
Entrega : Martes 26 de septiembre

Indicaciones

- La tarea es estrictamente individual.
- La solución debe ser entregada en el archivo `t1.py` del repositorio privado asignado mediante GitHub Classroom para esta tarea. Se revisará el último commit subido antes de la entrega al repositorio. Se usará Python 3.9.X para la revisión.
- El *input* para el programa debe ser obtenido desde *standard input*. El *output* debe ser entregado mediante *standard output*.
- La corrección se realizará mediante *tests* automatizados acordes al formato de *input* y *output* especificado. Cada *test* tendrá un timeout según lo que se especifica como tiempo esperado.
- Un *test* se considerará **reprobado** en caso de que 1) dado el *input* el *output* sea incorrecto, 2) exista un error de *runtime* durante su ejecución, o 3) el *timeout* se cumpla durante su ejecución. En otro caso, el *test* se considerará **aprobado**.
- No se permite el uso de librerías externas a la librería estándar de Python a *priori*. Consultar en las [issues del repositorio oficial del curso](#) en caso de requerir una.
- Para esta tarea aplica la política de atrasos descrita en el programa del curso.

Problema

Para un estudio estadístico se requiere el estudio de similitud de N secuencias de letras. Cada secuencia A_i es del mismo largo y necesita saber conocer el índice de similitud del conjunto de secuencias dada una relación \approx definida de forma recursiva. $A_i \approx A_j$ si y solo si se cumple uno de los siguientes casos:

- A_i es igual a A_j
- Si A_i y A_j tienen largo par, se pueden dividir y comparar los dos pares de mitades del mismo largo en cualquier orden, es decir, si A_{i1}, A_{i2} son las mitades de la primera secuencia y A_{j1}, A_{j2} las dos mitades de la segunda, alguno de los siguientes casos se cumple:
 - $A_{i1} \approx A_{j1} \wedge A_{i2} \approx A_{j2}$
 - $A_{i1} \approx A_{j2} \wedge A_{i2} \approx A_{j1}$

El índice de similitud es la cantidad de pares de secuencias distintas que cumplen la relación \approx :

$$\text{Índice de similitud} = \sum_{i=1}^n \sum_{j=i+1}^n F(A_i, A_j)$$

$$F(A_i, A_j) = \begin{cases} 1 & \text{si } A_i \approx A_j \\ 0 & \text{en otro caso} \end{cases}$$

Input

La primera línea contiene el entero N ($2 \leq N \leq 10$), indicando el número de secuencias. Las siguientes N líneas contienen las secuencias A_i ($1 \leq i \leq N$), cada una de largo $|A_i|$ ($1 \leq |A_i| \leq 10^5$), compuestas por letras minúsculas. Cada secuencia A_i es del mismo largo.

Output

El output debe ser un entero, el índice de similitud de las secuencias.

Tiempo esperado

Se espera que la solución se ejecute en un tiempo **menor o igual a 2 segundos** para cualquier instancia de input según las restricciones dadas.

Complejidad esperada

Se espera que la solución posea una complejidad de $O(N \cdot |A_i| \cdot \log(|A_i|))$.

Hint: Use dividir para conquistar. Para una cadena de caracteres cualquiera A , piense en como calcular la cadena de caracteres lexicográficamente menor B tal que $A \approx B$. Una cadena de caracteres B es menor lexicográficamente que A del mismo largo, si la primera posición en que difieren de izquierda a derecha es i , $B[i] < A[i]$ en orden alfabético.

Ejemplo

Los siguientes tests están ya cargados a GitHub Classroom con corrección automática mediante GitHub Actions. **Los tests para la corrección serán distintos a estos.**

Input de ejemplo 1
2 frff ffrf
Output de ejemplo 1
1

Input de ejemplo 2
2 abbabaab bbbaaaba
Output de ejemplo 2
0

Input de ejemplo 3
3 ccddffdd ddffddcc ffddddcc
Output de ejemplo 3
3