

Tarea 7

A continuación encontrará 3 preguntas, cada una dirá cuántos puntos vale en su preámbulo. Sea lo más detallado y preciso posible en sus razonamientos, algoritmos y demostraciones.

Además del informe expresando su solución, debe dar una implementación de su solución en el lenguaje de su elección (solamente como una función; el formato de entrada/salida no es relevante), para las preguntas 2 y 3.

La entrega se realizará únicamente por correo electrónico a rmonascal@gmail.com.

Fecha de entrega: Hasta las 11:59pm. VET del **Lunes, 11 de Marzo** (*Semana 9*).

1. (2 puntos) – Sea n una cadena de caracteres que tiene su número de carné (sin el guión)
 - (a) Construya un árbol de sufijos para n .
 - (b) Construya un arreglo de sufijos a partir del árbol de sufijos para n .
 - (c) Para cada posición k , calcule los valores para $PLCP[k]$ (el prefijo común permutado más largo) y $LCP[k]$ (el prefijo común más largo), como fue visto en clase.

2. (3 puntos) – Sea $P = \{p_1, p_2, \dots, p_n\}$ un conjunto de puntos en el plano cartesiano. Definimos una **capa** como aquellos puntos que forman parte del polígono convexo más pequeño que rodea a todos los puntos en P . Pero los puntos son como los ogros o las cebollas y pueden tener más de una capa. En particular, se puede remover la capa para P y obtener un conjunto de puntos P' a los que también se les puede calcular su capa. ¿Cuántas capas tiene el conjunto de puntos P ?

Diseñe un algoritmo que pueda responder a esta consulta usando tiempo $O(n^2 \log n)$ y memoria $O(n)$.

3. (4 puntos) – Considere una cadena de caracteres S , de longitud n . Se desea hallar la subcadena T de S más grande, tal que:

- T sea prefijo de S (la cadena S *empieza* con T)
- T sea sufijo de S (la cadena S *termina* con T)
- $T \neq S$

Considere los siguientes ejemplos:

- Para la cadena ABRACADABRA, la respuesta sería **ABRA**.
- Para la cadena AREPERA, la respuesta sería **A**.
- Para la cadena **ALGORITMO**, la respuesta sería λ (la cadena vacía).

Diseñe un algoritmo que pueda responder a esta consulta usando tiempo y memoria $O(n)$.