Práctica 10 - Problemas de búsqueda de subcadenas (string matching)

NOTAS PRELIMINARES
Los objetivos de esta práctica son:

- Comprender profundamente los algoritmos más importantes para el problema de búsqueda de subcadenas.
- Modificar esos algoritmos para situaciones particulares de la entrada o variantes del problema.

Los ejercicios marcados con el símbolo * constituyen un subconjunto mnimo de ejercitación. Sin embargo, aconsejamos fuertemente hacer todos los ejercicios.

Ejercicio 1 *

Supongamos un patrón P donde todos los caracteres sean diferentes. Mostrar como podra adaptarse el algoritmo *naive* (también llamado *de fuerza bruta*) para hallar en $O(n)$ las apariciones de P en un texto de n caracteres.

Ejercicio 2 *

Supongamos que permitimos que el patrón de búsqueda P contenga el carcter $*$ indicando que este puede ser reemplazado por 0 o más caracteres. Damos por hecho que $*$ no aparece en T.

Por ejemplo, el patrón $ab*ba*c$ ocurre en el texto $T = cabccbacbacab$, en $T[2, 8]$ y $T[2, 11]$ (siendo 1 la primera posición de T).

Dar un algoritmo *naive* pero polinomial para determinar si P se encuentra en T y analizar la complejidad del mismo.

Ejercicio 3 *

Cuántos falsos positivos encuentra el algoritmo de Rabin y Karp buscando el patrón $P = 26$ en el texto $T = 3141592653589793$, si se trabaja tomando módulo 11 en la función de *hash*?

Ejercicio 4 *

Cómo extendería el algoritmo de Rabin-Karp para hallar una ocurrencia de alguno de los k patrones pertenecientes a un conjunto de entrada?

Ayuda 1: Comenzar resolviendo el problema suponiendo que todos los patrones son de igual longitud y luego generalizar la solución al caso de longitudes mixtas.
Ayuda 2: $(a + b)mod(c) = ((amod(c)) + (bmod(c)))mod(c)$. *La misma propiedad vale para la multiplicación.*

Ejercicio 5 *

Dar un algoritmo para el problema del ejercicio **2** basado en el algoritmo del autómata. El tiempo de ejecución del mismo (sin contar el preprocesamiento) debe ser de $O(|T|)$.

Ejercicio 6 *

Dados dos patrones P y P', construir un autómata que permita hallar las apariciones de ambos patrones en un texto T. Se espera una solución donde se minimice la cantidad de estados.

Ejercicio 7 *

Dar un algoritmo $O(m|\Sigma|)$, donde m=$|P|$, para construir el autómata que permita buscar el patrón P en cualquier texto.

Ayuda: inspirarse en el algoritmo de Knuth, Morris y Pratt y demostrar que $\delta(q, a)$=$\delta(\pi[q], a)$ *si* q=m *o si* $P[q + 1]\neq a$.