

## Diccionario de Strings

💡 Aprovechamos que los objetos son strings.

→ Tries (árbol digital)

Cada nodo tiene 0 o más hijos y sus aristas están con un label. No pueden haber 2 aristas saliendo de un nodo con el mismo label.

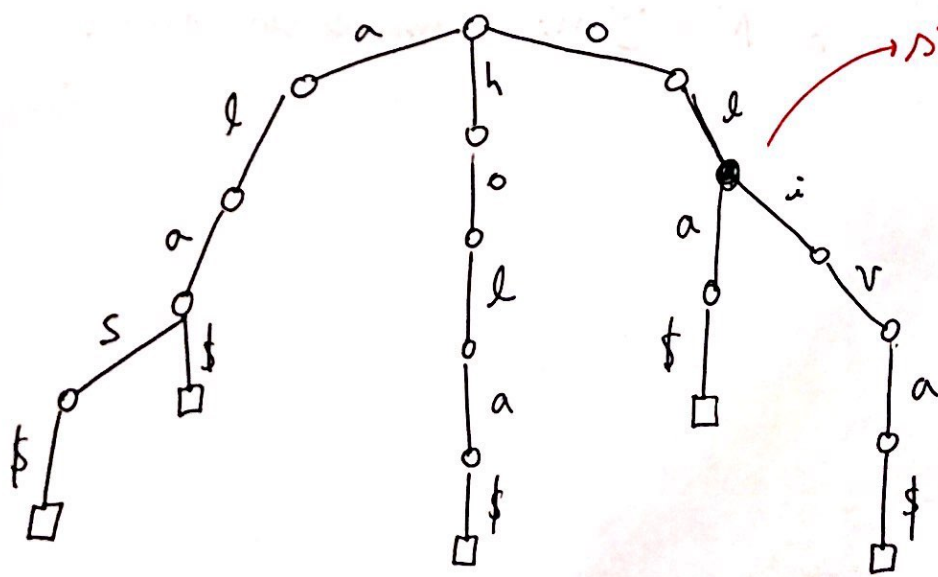
Al concatenar los labels del camino desde la raíz hasta la hoja, se obtiene un string.

la hoja, se obtiene un string.  
Todos los strings estarán terminados por el caracter "\$".

$str(v)$ : string que se obtiene desde la raíz hasta el nodo  $v$ .

$str(v)$ : string que se obtiene desde la raíz hasta  $v$ .  
 Si  $v$  es un nodo interno, entonces  $str(v)$  es un prefijo de algún string.

$S = \{ \text{hola, ola, oliva, ala, alas} \}$



str(r) = "ol"  
que es prefijo de  
"da, oliva"

# La raíz no tiene más hijos que strings en el alfabeto.

De esta forma, vamos buscando caracter por caracter y atravesando el árbol.

//Ejemplo:  $\text{buscar}("ala") \rightarrow S=[a, l, a]$   
 $S'=[a, l, a, \$]$  (añadimos \$)

Nos paramos en la raíz y buscamos la arista con el label  $S[0] = "a"$ . Así seguimos hasta encontrar la arista con el label "\$".

Si en algún caracter no encontramos la arista con ese label, significa que el string que buscamos no se encuentra en el trie.

La inserción y borrado están en el apunte, pero siguen la misma idea 😊

Tiempo  $O(m)$ , con  $m$  el largo del string

Memoria  $O(N)$ , con  $N = \sum m_i$  suma de los largos.

## → Árboles Patricia (blind trie)

espacio	$O(n)$
tiempo	$O(m)$

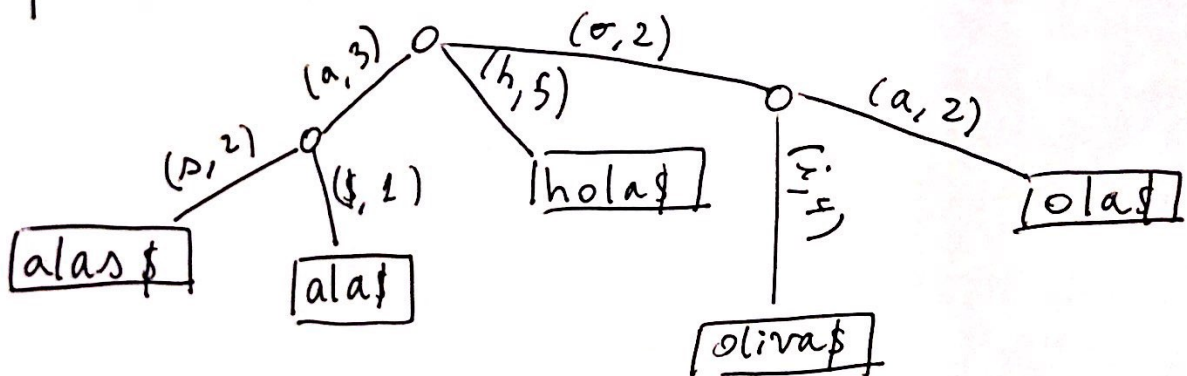
En este trie NO existen caminos unarios,  $\therefore$ , c/nodo

tiene al menos dos hijos y  $\exists$  n nodos hojas  $\Rightarrow$  total nodos  $O(n)$

Las aristas están rotuladas con un string, pero no almacena todo el string, si no solo su primer caracter a y el largo l  $\rightarrow (a, l)$

En este caso, para buscar un string S no vamos viendo caracter por caracter, sino por sub-strings.

// Ejemplo:



Buscar("asas") : 1) arista  $(a, 3)$  bajamos x ahí

2) Avanzamos 3 caracteres en  $S = [a, s, a, s, \$]$

3) Vemos qué arista comienza con  $S[3] = 's'$

4) Bajamos por  $(s, 1)$  y es una hoja

5) Si "asas" está en el trie, será en esta hoja.

6) ¿"asas\$" = "ala\$"?  $\Rightarrow$  NO