

# Análisis del String Matching y su optimización por medio de algoritmos existentes

Héctor Abraham Galván García

**Abstract**—En el presente reporte se examina el procedimiento para realizar empate de cadenas y su implementación mediante diversos algoritmos.

**Palabras clave:** String Matching, Cadena de caracteres, Optimización, Rabin-Karp, Patrón, Subcadena

## I. INTRODUCCIÓN

Un algoritmo de búsqueda de subcadenas tiene como objetivo encontrar una subcadena con alguna propiedad dada en una cadena de caracteres. Se le denomina patrón a la subcadena buscada y texto a la cadena en la que se realiza la búsqueda. Dada su naturaleza, la longitud del texto es mayor a la del patrón. Existen varios algoritmos de búsqueda de subcadenas, pero el reporte se centrará en el empate de cadenas simple (sólo se busca una subcadena) por medio de la fuerza bruta y del algoritmo de Rabin-Karp.

## II. FUERZA BRUTA

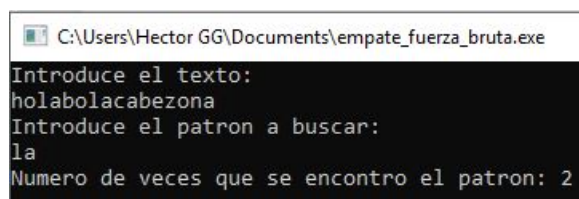
Para un algoritmo de empate de cadenas que se basa en el método de fuerza bruta, se recorre el patrón sobre el texto de izquierda a derecha, comparándolo con las subcadenas del mismo tamaño que empiezan en cada carácter del texto. En otras palabras, recorremos el texto comparándolo con el patrón en su primera posición hasta que se encuentre una coincidencia, repitiendo el proceso con cada carácter del patrón hasta que se termine la coincidencia o termine el patrón.

## III. RABIN KARP

Para este algoritmo de búsqueda de subcadenas, se trata cada uno de los grupos de  $m$  caracteres del texto ( $m$  es el número de caracteres en el patrón) como un índice en una tabla de valores hash. Si coincide la función con el patrón, tal vez se encontró una coincidencia y hay que verificarlo comparando el texto con el patrón por fuerza bruta.

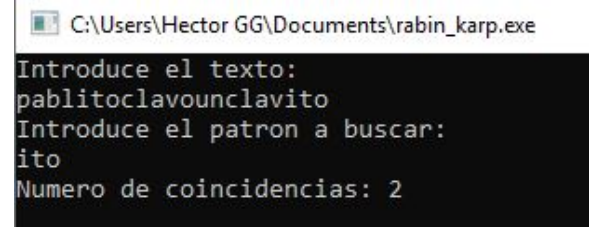
Para el tamaño de la tabla de dispersión se tiene un número primo, que puede ser dado o calculado, dependiendo de las necesidades. Para el recorrido de las cadenas se tomará  $N$  como el número de caracteres del texto.

## IV. PRUEBAS



```
C:\Users\Hector GG\Documents\empate_fuerza_bruta.exe
Introduce el texto:
holabolacabezona
Introduce el patron a buscar:
la
Numero de veces que se encontro el patron: 2
```

Fig. 1. Fuerza bruta



```
C:\Users\Hector GG\Documents\rabin_karp.exe
Introduce el texto:
pablito clavounclavito
Introduce el patron a buscar:
ito
Numero de coincidencias: 2
```

Fig. 2. Rabin-Karp

## V. CONCLUSIONES

Cuando se trata del String Matching, no se puede ignorar el concepto de optimización, porque aunque para fines de aprendizaje no se revisen cadenas muy largas, cuando se trata de realizar una verdadera búsqueda, el tiempo de respuesta en un algoritmo de pura fuerza bruta se elevará bastante. Por eso es importante tener algoritmos de búsqueda que optimicen ese tiempo de respuesta.