



INSTITUTO POLITECNICO NACIONAL
UNIDAD PROFESIONAL INTERDISCIPLINARIA DE
INGENIERIA CAMPUS ZACATECAS



Alumno:

Mariel López Beltrán

Docente:

Roberto Oswaldo Cruz Leija

Grupo:

3CM1

Asignatura:

Análisis de algoritmos

Tarea:

Rabin Karp

Fecha de entrega:

14/11/2019

Introducción:

El algoritmo de Rabin-Karp es un algoritmo de búsqueda de cadenas creado por Richard M. Karp y Michael O. Rabin que utiliza el hash para encontrar cualquiera de un conjunto de cadenas de patrones en un texto.

En el presente reporte se llevara a cabo la explicación y el funcionamiento del algoritmo Rabin-Karp de acuerdo a los pasos que este lleva a cabo para la resolución de las cadenas que se ingresen y dependiendo del patrón este determinara si el patrón se encuentra dentro de la cadena o no, en caso de ser cierto este algoritmo mostrara el número de veces en que aparece el patrón.

Marco Teórico:

El algoritmo Rabin-Karp es un algoritmo de búsqueda de cadenas que utiliza el hash para encontrar patrones en las cadenas. Una cadena es un tipo de datos abstracto que consiste en una secuencia de caracteres. Las letras, palabras, oraciones y más se pueden representar como cadenas.

La coincidencia de cadenas es una aplicación informática muy importante. Si alguna vez buscó en un documento una palabra en particular, entonces se ha beneficiado de la tecnología de coincidencia de cadenas. La coincidencia de cadenas también se puede utilizar para detectar plagio comparando cadenas en el documento AA con cadenas en el documento BB.

```
Procedure Rabin-Karp(Text, Pattern, Prime):  
  m := Pattern.Length  
  HashValue := Calculate-Hash(Pattern, Prime, m)  
  CurrValue := Calculate-Hash(Pattern, Prime, m)  
  for i from 1 to Text.length - m  
    if HashValue == CurrValue and String-Match(Text, Pattern, i) is true  
      Return i  
    end if  
    CurrValue := Recalculate-Hash(String, i+1, Prime, CurrValue)  
  end for  
  Return -1
```

Este algoritmo se utiliza en la detección de plagio. Dado el material de origen, el algoritmo puede buscar rápidamente en un papel ejemplos de oraciones del material de origen, ignorando detalles como el caso y la puntuación. Debido a la abundancia de las cadenas buscadas, los algoritmos de búsqueda de una sola cadena no son prácticos aquí. De nuevo, el algoritmo Knuth-Morris-Pratt o el algoritmo de búsqueda de cadenas de Boyer-Moore es un algoritmo de búsqueda de cadenas de un solo patrón más rápido que Rabin-Karp. Sin embargo, es un algoritmo de elección para la búsqueda de múltiples patrones. Si queremos encontrar alguno de los números grandes, digamos k , patrones de longitud fija en un texto, podemos crear una variante simple del algoritmo de Rabin-Karp.

Para el texto de longitud N y P patrones de longitud combinada m , su promedio y mejor de los casos tiempo de ejecución es $O(n + m)$ en $O(p)$ espacio, pero su tiempo del peor caso es $O(nm)$.

Características de programación:

- Calcular el valor hash del patrón
- Calcular el valor hash de los primeros M caracteres del texto
- Compare ambos valores hash
- Si no son iguales, calcule el valor hash para los siguientes M caracteres de texto y compare nuevamente.
- Si son iguales, realice una comparación de fuerza bruta.

Pruebas de ejecución:

```
E:\Rabin_karp\main.exe
RABIN KARP
Cadena ingresada: ABAAABCDBBABCDDDEBCABC
Patron a encontrar A
se ha encontrado en la posicion :0
se ha encontrado en la posicion :2
se ha encontrado en la posicion :3
se ha encontrado en la posicion :4
se ha encontrado en la posicion :10
se ha encontrado en la posicion :18
numeros repeticion :6
```

```
E:\Rabin_karp\main.exe
RABIN KARP
Cadena ingresada: ABAAABCDBBABCDDDEBCABC
Patron a encontrar CD
se ha encontrado en la posicion :6
se ha encontrado en la posicion :12
numeros repeticion :2
```

```
E:\Rabin_karp\main.exe
RABIN KARP
Cadena ingresada: ABASDFGHJKLASDFGHJKASDFGHJKASDFGHJK
Patron a encontrar AS
se ha encontrado en la posicion :2
se ha encontrado en la posicion :11
se ha encontrado en la posicion :19
se ha encontrado en la posicion :27
numeros repeticion :4
```

Conclusiones:

De acuerdo al presente reporte se llevó a cabo la implementación de dicho algoritmo, como se muestra en las pruebas de ejecución dependiendo de la cadena que se ingrese y el patrón que requiera encontrar, cabe destacar que dependiendo de la cadena este la va partiendo y de esta forma de manera matemática va sacando el valor de cada una de las partes de la cadena , sin embargo, va restando para no sacar el valor de cada una de las partes y de esta forma puede encontrar el patrón requerido de una forma rápida