INSTITUTPO POLITÉCNICO NACIONAL

U P I I Z

Ing. Sistemas Computacionales

Mtro. Roberto Oswaldo Cruz Leija

Análisis de Algoritmos

Rabin Karp

Jazmín Abigail Mena Zamora

Es un Algoritmo de búsqueda de subcadenas simple enunciado por Michael Oser Rabin y Richard Manning Karp en 1987.1 Este algoritmo se basa en tratar cada uno de los grupos de m caracteres del texto (siendo m el número de símbolos del patrón) del texto como un índice de una tabla de valores hash (la llamaremos tabla de dispersión), de manera que si la función hash de los m caracteres del texto coincide con la del patrón es posible que hayamos encontrado un acierto. para verificarlo hay que comparar el texto con el patrón, ya que la función hash elegida puede presentar colisiones.

La función hash tiene la forma d(k)=xk mod d donde d es un número primo grande que será el tamaño de la tabla de dispersión y xk se calcula de la forma indicada más abajo.

Para transformar cada subcadena de m caracteres en un entero lo que hacemos es representar los caracteres en una base B que en el planteamiento original coincide con el tamaño del alfabeto. Por tanto el entero xi correspondiente a la subcadena de texto Ci...Ci+m−1 sería:

$$xi=Ci\times Bm-1+Ci-1\times Bm-2+...+Ci+m-1$$

podemos calcular el valor de xi+1 en función de xi

$$xi+1=xi\times B-Ci\times Bm+Ci+m$$

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** espacio **(p),** pero su tiempo del peor caso es **O (nm).**

Resultados:

Caso 1

```
char *ejemplo, *patron;
int num= 11; //es necesario el num primo
clrscr();
char ejemplo[] = "holacaradebola";
char patron[] = "bola";
```

```
se ha encontrado en la posicion :10
numeros repeticion :1
```

Caso 2:

```
    char *ejemplo, *patron;
    int num= 11; //es necesario el num primo
    clrscr();
    char ejemplo[] = "holacaradebola";
    char patron[] = "hey";
```

```
numeros repeticion :0
```

Conclusiones:

Este es un algoritmo que nos calcula un hash para así poder comparar palabras que se vayan introduciendo, a cada unidad de comparación se da la una palabra clave, de esta manera cuando se ocupe comparar los ejemplos dados, solo se camapara entre sus claves y no en letras (lo que lo hace más fácil y rápido). Es un muy buen algoritmo para saber si un documento es igual o qué tanto porcentaje tiene de igual.

La única desventaja que tiene es que el sistema no puede saber cuál fue el primer o segundo documento. Sólo sabe si sí son iguales o similares los documentos.