NAME:	OMKAR DESHMUKH
UID:	2021700018
DIV:	D1
EXP:	10
AIM:	String Matching algorithms
PROBLEM STATEMENT:	To implement Robin Karp algorithm
THEORY/ALGORITHM:	The Robin-Karp algorithm is a string matching algorithm that uses a circular hash function to quickly find occurrences of a pattern string in a larger text string. The algorithm works by computing a hash value for the pattern and then comparing the hash values of substrings of the text to the hash value of the pattern. The algorithm works as follows: 1. Compute the hash value of the pattern string. 2. Compute the hash value of the first substring of the text string with the same length as the pattern. 3. Compare the hash value of the substring with the hash value of the pattern. If they are equal, check if the substring and pattern are actually equal. 4. If the hash values are not equal, move the sliding window one position to the right and compute the hash value of the new substring by subtracting the value of the character leaving the window and adding the value of the character entering the window. Then repeat step 3. Time Complexity: O(n+m); where n is the length of text and m is the length of the pattern The hash function used in the Robin-Karp algorithm is a circular hash function that updates the hash value of a substring by subtracting the value of the first character and adding the value of the last character. This allows the algorithm to compute the hash value of each substring in O(1) time.
CODE:	#include <stdio.h> #include <string.h>

```
void
matchString (char str[], char pat[], int q)
{

int x = strlen (pat);
  int y = strlen (str);

int p = 0, t = 0, h = 1, d = 256;
  int i, j;

for (i = 0; i < x - 1; i++)

   {

h = (h * d) % q;

}

for (i = 0; i < x; i++)

   {

p = (d * p + pat[i]) % q;
     t = (d * t + str[i]) % q;

}

for (i = 0; i <= y - x; i++)

   {

if (p == t)

         {

for (j = 0; j < x; j++)

              {

if (str[i + j] != pat[j])
```

```
                {
                  break;

}
            }
           if (j == x)

            {

printf ("\nMatch Found at index: %d \n", i);
              return;

}
         }

if (i < y - x)

         {

t = (d * (t - str[i] * h) + str[i + x]) % q;

if (t < 0)
{
   t=(t+q);
}
}
}
printf("Match is not found");
}

int main(){
   char str[100],pat[100];
   printf("string:");
   scanf("%[^\n]s",str);
   getchar();
   printf("pattern:");
    scanf("%[^\n]s",pat);
    int q=269;
    matchString(str,pat,q);
```

	return 0; }
OUTPUT:	
CONCLUSION:	FROM THIS EXPERIMENT, I HAVE LEARNED ROBIN KARP ALGORITHM AND CONCEPTS RELATED TO IT.