Name	**Harsh Chandra**
UID no.	**2021700013**
Experiment No.	**10**

AIM:	**String Matching algorithms (To implement Robin Karp algorithm)**
Program 1	
ALGORITHM/ THEORY:	Rabin Karp algorithm is based on hashing technique. It first computes the hash value of p and t_s. • If hash values are same, i.e. if hash(p) = hash(t_s), we check the equality of inverse hash similar to a naïve method. If hash values are not same, no need to compare actual string. • On the hash match, actual characters of both strings are compared using brute force approach. If the pattern is found, then it is called *hit*. Otherwise, it is called a *spurious hit*.

PROGRAM:	**#include <stdio.h>
	#include <string.h>
	
	#define d 256
	
	void search(char pat[], char txt[], int q)
	{
	int M = strlen(pat);
	int N = strlen(txt);
	int i, j;
	int p = 0; // hash value for pattern
	int t = 0; // hash value for txt
	int h = 1;
	
	for (i = 0; i < M - 1; i++)
	h = (h * d) % q;**

```
        for (i = 0; i < M; i++) {
                p = (d * p + pat[i]) % q;
                t = (d * t + txt[i]) % q;
        }

        for (i = 0; i <= N - M; i++) {

                if (p == t) {
                        for (j = 0; j < M; j++) {
                                if (txt[i + j] != pat[j])
                                        break;
                        }

                        // if p == t and pat[0...M-1] = txt[i, i+1,
                        // ...i+M-1]
                        if (j == M)
                                printf("Pattern found at index %d \n", i);
                }

                // Calculate hash value for next window of text:
                // Remove leading digit, add trailing digit
                if (i < N - M) {
                        t = (d * (t - txt[i] * h) + txt[i + M]) % q;

                        if (t < 0)
                                t = (t + q);
                }
        }
}

int main()
{
        char txt[50];
        char pat[50];
        printf("Enter txt string.\n");
   scanf("%s", txt);

   printf("Enter pattern string.\n");
   scanf("%s", pat);
```

```
        int q = 101;
        search(pat, txt, q);
        return 0;
}
```

RESULT:

```
Enter txt string.
AABAACAADAABAABA
Enter pattern string.
AABA
Pattern found at index 0
Pattern found at index 9
Pattern found at index 12


...Program finished with exit code 0
Press ENTER to exit console.
```

CONCLUSION: From this experiment, I understood how to implement string matching algorithm Rabin Karp algorithm to search pattern in a string.