School of Computer Science Engineering and Technology

Course- B. Tech | Type- Core Course

Code- CSET206 | Course Name-DAA

Year- 2023 | Semester- Even

Date- 05/04/2023 | Batch- 4th Sem

Tutorial: 11

Objective 1: Understand the Rabin Karp Pattern matching algorithm

Tut No.	Name	CO 1	CO 2	CO 3
11.3	Tutorial:11	√		

Rabin Karp String Matching Algorithm

Objective: If we have text string **T** and pattern string **P**, we want to determine whether or not P is found in T, i.e., P is a substring of T.

Algorithm:

1. Hash P to get h(P)
2. Iterate through all length L substrings of S, hashing those substrings and comparing to h(P)
3. If a substring hash value does match h(P), do a string comparison on that substring and P, stopping if they do match and continuing if they do not.

Example: Given T = 31415926535 and P = 26

We choose q = 11

h(P)=P mod q = 26 mod 11 = 4

3	1	4	1	5	9	2	6	5	3	5

31 mod 11 = 9 ≠ 4 (No match)

3	1	4	1	5	9	2	6	5	3	5

14 mod 11=3≠ 4 (No match)

3	1	4	1	5	9	2	6	5	3	5

41 mod 11= 8 ≠ 4 (No match)

3	1	4	1	5	9	2	6	5	3	5

15 mod 11 =4 but 26 ≠ 15 so it is a spurious hit

3	1	4	1	5	9	2	6	5	3	5

59 mod 11 = 4 but 59≠ 26 so it is a spurious hit

3	1	4	1	5	9	2	6	5	3	5

92 mod 11 = 4 but 92 ≠ 26 so it is a spurious hit

3	1	4	1	5	9	2	6	5	3	5

26 mod 11 = 4 it is an **exact match**

3	1	4	1	5	9	2	6	5	3	5

65 mod 11= 10≠ 4 (No match)

3	1	4	1	5	9	2	6	5	3	5

53 mod 11 = 9 ≠ 4 (No match)

3	1	4	1	5	9	2	6	5	3	5

35 mod 11 = 2 ≠ 4 (No match)

Problems to Ponder

Problem 1:

Working modulo q = 17, how many spurious hits does the Rabin-Karp matcher encounter in the text T = 23590231415267399 21 when looking for the pattern P = 7399?

Problem 2:

Working modulo q = 11, how many spurious hits does the Rabin-Karp matcher encounter in the text T = 3151429563589793 when looking for the pattern P = 589

Problem 3:

What is the time complexity of Rabin Karp string matching algorithm where length of T = n and length of P =m?

Problem 4:

Suggest a good Hash function to quickly compute the hash value of substring of text T corresponding to pattern P. (**Hint**: use a Rolling Hash Function)