Advanced Algorithms & Data Structures

Department of CSE

ADVANCED ALGORITHMS AND DATA STRUCTURES
23CS03HF

Topic:

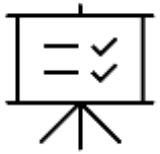
Rabin Karp String Matching Algorithnm

Session - 31

AIM OF THE SESSION

To familiarize students with the concept of Rabin-Karp string matching algorithm.

INSTRUCTIONAL OBJECTIVES



This Session is designed to:

1. Demonstrate :- Rabin Karp string matching algorithm.
2. Describe :- Sequence of steps in Rabin Karp string matching algorithm

LEARNING OUTCOMES

At the end of this session, you should be able to:

1. Define :- Rabin Karp string matching algorithm.
2. Describe :- Sequence of steps in Rabin Karp string matching algorithm
3. Summarize:- Identification of patterns using Rabin Karp string matching algorithm

Rabin-Karp string matching algorithm

- Rabin and Karp proposed a string-matching algorithm to find the pattern in a more efficient way.
- It also checks the pattern by moving window one by one, but without checking all characters for all cases, it finds the hash value.
- When the hash value is matched, then only it tries to check each character.
- This procedure makes the algorithm more efficient compared to naïve string matching algorithm.

How Rabin-Karp works

- Given a pattern P [1..m],
- let p denote its corresponding hash value and
- a text T [1..n],
- let t_S denote the hash value of the length-m substring T
- T[s+1..s+m],for s=0,1..n-m.
- Certainly, s is a valid shift iff t_S == p and T[s+1..s+m] == P[1..m] otherwise s is a invalid shift.

- Let characters in both arrays T and P be digits in radix-S notation. (S = (0,1,...,9)
- Choose a prime number *q* such that fits within a computer word to speed computations.
- Compute (p mod q)
  - The value of p mod q is what we will be using to find all matches of the pattern P in T.

Rabin-Karp : Algorithm

RABIN-KARP-MATCHER(*T, P, d, q*)

1. $n = length[T]$
2. $m = length[P]$ (q is a prime number)
3. $h = d^{m-1} \bmod q$
4. $p = 0$
5. $t_0 = 0$
6. **for** $i = 1$ **to** m **do**
7. $p = (dp + P[i]) \bmod q$
8. $t_0 = (dt_0 + T[i]) \bmod q$

for s =0 **to** $n - m$ **do**

10. **if** $p == t_s$ **then**
11. **if** $P[1 . . m] = T[s + 1 . . s + m]$ **then**
12. **Print** "Pattern occurs with shift" s
13. **if** $s < n - m$ **then**
14. $t_{s+1} = (d(t_s - T[s + 1]h) + T[s + m + 1]) \bmod$

Rabin-Karp Example

- Given T = 31415926535 and P = 26
- We choose q = 11
- P mod q = 26 mod 11 = 4

31 mod 11 = 9 not equal to 4

14 mod 11 = 3 not equal to 4

41 mod 11 = 8 not equal to 4

15 mod 11 = 4 equal to 4 -> spurious hit

59 mod 11 = 4 equal to 4 -> spurious hit

92 mod 11 = 4 equal to 4 -> spurious hit

26 mod 11 = 4 equal to 4 -> an exact match!!

65 mod 11 = 10 not equal to 4

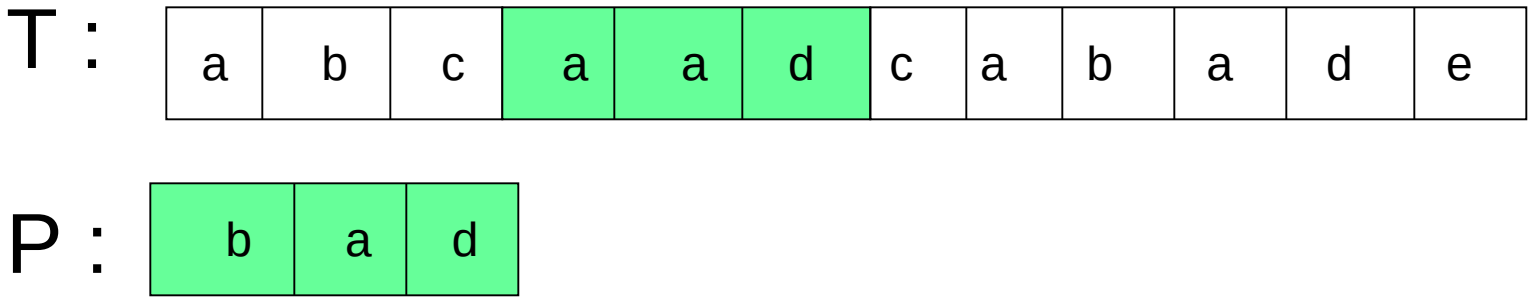
53 mod 11 = 9 not equal to 4

35 mod 11 = 2 not equal to 4

As we can see, when a match is found, further testing is done to insure that a match has indeed been found.

Time Complexity

- The running time of the Rabin-Karp algorithm in the worst-case scenario is O(n-m+1)m but it has a good average-case running time.
- If the expected number of valid shifts is small O(1) and the prime q is chosen to be quite large, then the Rabin-Karp algorithm can be expected to run in time O(n+m) plus the time to required to process spurious hits.



FIND WHETHER PATTERN EXISTS IN THE TEXT OR NOT USING ROBIN-KARP STRING MATCHING ALGORITHM

SUMMARY

- The **Rabin-Karp** algorithm is a string-searching algorithm that uses hashing to find patterns in strings.

SELF-ASSESSMENT QUESTIONS

The Rabin-Karp algorithm primarily uses which of the following techniques to compare substrings?

(a) Dynamic Programming
(b) Hashing
(c) Divide and Conquer
(d) Backtracking

In the Rabin-Karp algorithm, what is the purpose of using a modulus operation in the hash function?

(a) To reduce the size of the hash value to a manageable number
(b) To ensure the hash value is always positive
(c) To avoid collisions in hash values
(d) To increase the computational complexity of the hash function

TERMINAL QUESTIONS

1. In the Rabin-Karp algorithm, how is the hash value of a substring computed and updated efficiently as the algorithm progresses?
2. What are the primary advantages and disadvantages of using the Rabin-Karp algorithm for string matching?

REFERENCES FOR FURTHER LEARNING OF THE SESSION

Reference Books :

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein., 3rd, 2009, The MIT Press.
2 Algorithm Design Manual, Steven S. Skiena., 2nd, 2008, Springer.
3 Data Structures and Algorithms in Python, Michael T. Goodrich, Roberto Tamassia, and Michael H. Goldwasser., 2nd, 2013, Wiley.
4 The Art of Computer Programming, Donald E. Knuth, 3rd, 1997, Addison-Wesley Professiona.

MOOCS :

1. https://www.coursera.org/specializations/algorithms?=
2.https://www.coursera.org/learn/dynamic-programming-greedy-algorithms#modules

THANK YOU