The Rabin-Karp Algorithm

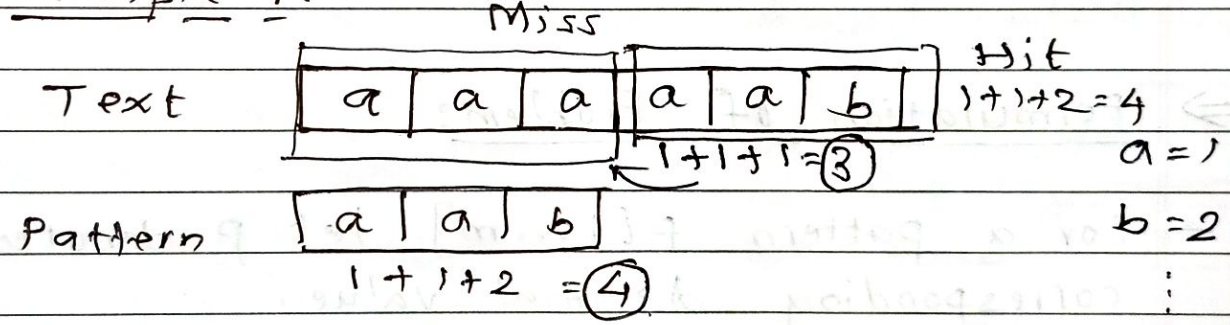
→ This algorithm makes use of elementary number theoretic notion such as "the equivalence of two numbers modulo a third number"

→ Assumption:

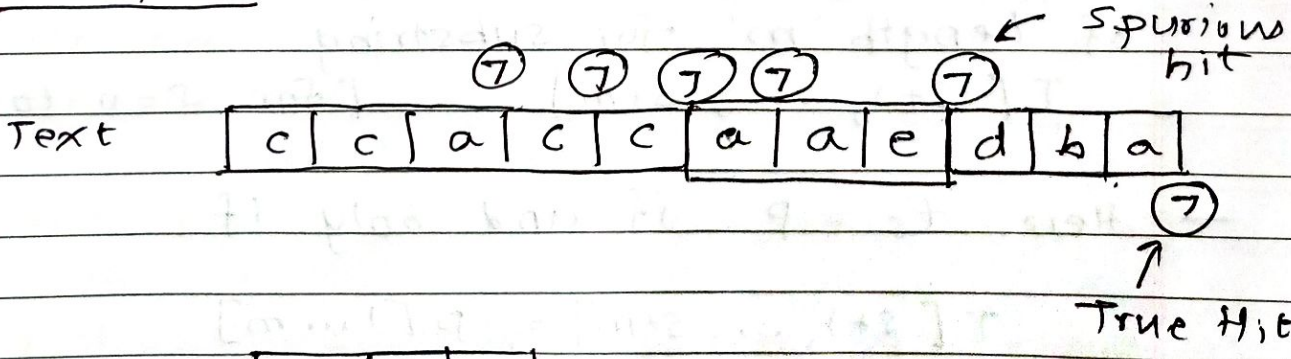
- $\Sigma = \{0, 1, \ldots, 9\}$ i.e. Each character is digit.
- In general, each character is a digit in radix-d notations
- As per assumption, we can view a string of 'k' consecutive numbers as length 'k' decimal number.

e.g. '31415' is treated as number 31415

Example 1:

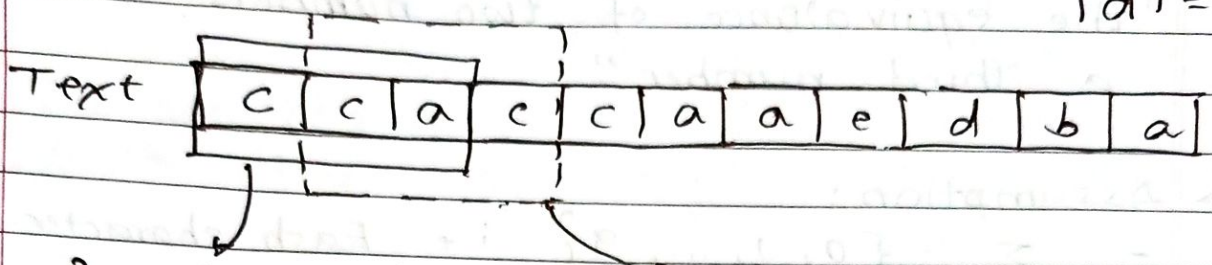


Example 2:



Example 3: (Rolling Hash function)

$|d| = 10$



$3 \times 10^2 + 3 \times 10^1 + 1 \times 10^0 = 331$

$3 \times 10^2 + 1 \times 10^1 + 3 = 313$

$[[331 - 3 \times 10^2] * 10] + 3 = 313$

a = 1
b = 2
c = 3
d = 4
e = 5

Pattern : d b a

$4 \times 10^2 + 2 \times 10^1 + 1 = 421$

⇒ Formulation of Problem:

- For a pattern $p[1 \ldots m]$ let p denotes corresponding decimal value.
- For a text $T[1 \ldots n]$
  Let t_s denotes the decimal value of length 'm' for substring
  $T[s+1, \ldots, s+m]$ ... [for $s = 0$ to $n-m$]

→ Here $t_s = p$ if and only if

$T[s+1, \ldots, s+m] = p[1 \ldots m]$

∴ s is valid shift iff $t_s = p$

- P can be computed in $\Theta(m)$ times using Horner's Rule.

$$p = P[m] + 10(P[m-1] + 10[P[m-2] + \ldots + 10(P[2] + 10(P[1])))$$

$$= P[m] + 10(P[m-1]) + 100(P[m-2]) + \ldots + 10^{m-1} P[1]$$

- Similarly, t_0 can also be computed.

- To compute remaining $t_1 \ldots t_{n-m}$ values in $\Theta(n-m)$, Rolling Hash Function can be used.

$$t_{s+1} = 10(t_s - 10^{m-1} T[s+1]) + T[s+m+1]$$

Eg. $m = 5$ $\quad t_s = 31415$ $\quad t_{s+1} = 2$

$$t_2 = 10(31415 - 10^4 \times 3) + 2$$
$$= 14150 + 2$$
$$= 14152$$

- We can compute t_{s+1} from t_s in constant time.

⇒ But if p and t_s are very large numbers then we can't assume that each arithmetic operation of p will take constant time.

- To overcome this problem, p and t_s values are computed with suitable modulo q function.
- If we choose the modulo q as prime such that $10q$ fits one computer word.
- In general, with d-array alphabet $\{0, 1, 2, \ldots, d-1\}$ choose q such that dq fits within one computer word.

→ General formula for calculating t_{s+1} from t_s will be

$$t_{s+1} = \left(d(t_s - T[s+1]h) + T[s+m+1]\right) \bmod q$$

where $h = d^{m-1} \pmod q$

It is the value of digit '1' in the higher order position

→ The solution of working with modulo q is not perfect because

(i) $t_s \equiv p \pmod q$ does not imply $t_s = p$

(ii) If $t_s \neq p \pmod q$ then definitely $t_s \neq p$ so the shift is invalid

→ So test is used to rule out invalid shift

⇒ If test is equal then it is further tested to check whether it is spurious hit or not.

– Additional test explicitly checks condition
$p[1...m] = T[s+1, ..., s+m]$

⟹ Algorithm:

Input : Text T, Pattern P, radix d (|Σ|)
Prime Number q

```
Rabin_karp ( T, P, d, q )
{
1.   n = T.length
2.   m = P.length
3.   h = d^(m-1) mod q
4    p = 0
5    t0 = 0
6.   for i=1 to m          do
7.        p = (d.p + p[i]) mod q
8.        t0 = (d.t0 + T[i]) mod q
9.   for s=0 to n-m        do
10.       if p==ts
          {
11.           if p[1...m] = T[s+1 ... s+m]
              {
12.               Print ("Pattern occur at
                          shift "s")
              }
13.       if s < n-m then
14.           t(s+1) = (d(ts - T[s+1]h) + T[s+m+1])
                       mod q
}
```

⇒ Complexity:
- Preprocessing Time: $\Theta(m)$
- Matching Time: $\Theta((n-m+1)\cdot m)$
  in worst case.
  Algorithm verifies all shifts explicitly
- In many application, there will be few valid shifts (say valid 'c' shifts)

→ Expected matching time $\Theta(n-m+1) + cm$
$\approx \Theta(n+m)$ + spurious hits processing

Q. Pattern = 31415 ⇒ m = 5 n = 10
Text = 9023141 52

d = 10
q = 13

Solution: Calculate p = 7 (Calculation of p^ at last shown)
Calculate $h = d^{m-1} \bmod q$
$= 10^4 \bmod 13$
$= 3$

Shift	Comparison	Hash Value	Result
0	90231 4152 31415	$t_0 = 9$ $= 90 \% 13 = 12$ $= 122 \% 13 = 5$ $= 53 \% 13 = 1$ $= 11 \% 13 = 11$ $t_0 = 11$	Invalid
1	9 02314 152 31415	$t_1 = [10(11 - 9(3)) + 4] \% 13$ $= [10 \times (-16) + 4] \% 13$ $= -156 \% 13$ $= 0$	Invalid

Shift	Comparison	Hash Value	Result
2	90[23141]52 31415	$t_2 = [10(0 - 0(3)) + 1] \% 13$ $= 1$	Invalid
3	902[31415]2 31415	$t_3 = [10(1 - 2(3)) + 5] \% 13$ $= (-50 + 5) \% 13$ $= -45 \% 13$ $= -6$ $t_3 = -6 + 13 = 7$	Valid
4	9023[14152] 31415	$t_4 = [10(7 - 3(3)) + 2] \% 13$ $= (-20 + 2) \% 13$ $= -18 \% 13$ $= -5$ $t_4 = -5 + 13 = 8$	Invalid

Calculation of P

$P = 3$

$= [3 \times 10 + 1] \% 13 = 5$

$= [5 \times 10 + 4] \% 13 = 2$

$= [2 \times 10 + 1] \% 13 = 8$

$= [8 \times 10 + 5] \% 13 = 7$

$\boxed{P = 7}$