★ Rabin karp string matching algorithm

It is an algorithm that uses hashing to find patterns in strings. let n be the length of the Text T and m be the length of the Pattern P. We need to find at which all positions in T, Pattern P exists.

⇒ Rabin karp method calculates hash value of pattern P.

⇒ It also calculates hash value for a substring of length m from T

⇒ If hash values are unequal algorithm will ~~compare~~ calculate the hash value for next m character sequence

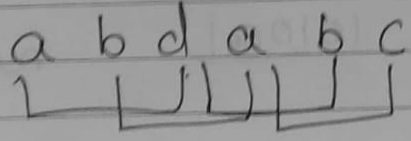
⇒ if the hash values are equal, algorithm will Compare the pattern and the

m character sequence.

⇒ In this way there is only one comparison per text subsequence & character matching is only needed when hash values match.

text = abdabc
pattern = abc

h(abc) = x



hash(abd) = y_1

Is $y_1 = x$?

No

hash(bda) = x

Is x = x ?

⇩

yes

now compare each character since there is mismatch take next substring

hash(dab) = y_2

Is $y_2 = x$?

No

hash(abc) = x

x = x ?

yes

Compare each character now. all characters are matching, so pattern is found at index 3.

Now, How to efficiently generate hash values is a question.

We are using Rolling hash function
we are considering
Prime = 3 to keep calculations simple at this moment - but in reality we should use higher value for prime such as 111 and so on..

1) x = old hash - val (old char)
2) x = x / Prime
3) newhash = x + Prime^{m-1} * val(newChar)

a → 1
b → 2
c → 3
d → 4
e → 5

g → 26

we have assumed these values to keep calculations simple but in reality you should choose ascII values of a, b, c---

Consider T = abeda

a b e
$1 + 2 \times 3^1 + 5 \times 3^2 = 52$

while calculating hash for
b e d
$2 + 5 \times 3^1 + 4 \times 3^2 = 2 + 15 + 36 =$
since now a is leaving we subtract 'a' value from 53
i.e 53 - 1 = 52/3
length of P - 1 ↓ (2)
= 17 + 4 (value of new char) × 3

$= 17 + 36$

$= 53$

So, In 3 steps we calculate hash & for bed from hash of abe

hash(eda)

e d a

$5 + 4 \times 3^1 + 1 \times 3^2 = 26$

hash(eda) b

$53 - 2 = 51$

$51/3 = 17 + 1 \times 3^2$

$= 17 + 9$

$= 26$

Subtract b value

$26 - 2$

$= 24$

divide by 3

$= 24/3$

$= 8 + 1 \times 3^2$

new character

$= 8 + 9$

$= 26$

example: Pattern = abc

text = abedabc

$h(abc) = 1 + 2 \times 3^1 + 3 \times 3^2$

$= 1 + 6 + 27$

$= 34$ ← pattern hash value

Now take substring abe

$hash(abe) = 1 + 2 \times 3^1 + 5 \times 3^2$

$= 1 + 6 + 45$

$= 52$

$52 \neq 34$

Now roll forward

hash(bed) = (52-1)/3 = 17+4×3^2

= 17+36

= 53

53 ≠ 34

Now roll forward

hash(eda) = 53-2 = 51/3 = 17+1×3^2

= 26

26 ≠ 34

roll forward

hash(dab) = 26-5

= 21/3

= 7+2×3^2

= 7+18

= 25

25 ≠ 34

roll forward

hash(abc) = 25 - 4 = 21/3 = 7+3×3^2

= 7+27

= 34

34 == 34 ?

Now Compare each Character of the pattern & substring. if all characters are equal return Index.

Time Complexity in worst case O(mn)

Applications - ① plagarism check.

② if multiple patterns are to be searched in a string Rabin Karp is useful.

text = abcgabcflmxyz

pattern1 = gab = hash(gab) = x1
pattern2 = xyz = hash(xyz) = x2
pattern3 = abc = hash(abc) = x3

Calculate hash for each substring in a text. If it matches with any of the 3 hash values characters from both are compared & if matching is succesful we can say that pattern is present in a Text t.