String Matching

→ Why its imp
- ↳ Word processors ↳ Crawlers
- ↳ Search Engine
- ↳ medical field (DNA matching) / (takes hours / days)
- ↳ Database Searches
- ↳ Compilers (Syntax / Parsing)

2 Types : Exact String Matching / Approximate String matching
↓ defined number of errors allowed ; mispiled misspelled

1. → Naive String matching (Brute force)
2. → Rabin Karp (RK Algo).
3. → Finite Automata
4. → KMP → Knuth - Morris - Pratt Algo

⇒ Naive String Matching (Brute force).

Text. length = N
pattern. length = m.

```
for ( i=0 to n-m ; i++ )
    if ( p(0 + m) = T( i to i+n)
        found.
```

e.g

T = a b cd a b b c x y z p q

P = a b b c.

((n - m+1) m) naive

No preprocessing

Notations

Σ^* = set of all finite strings from alphabet Σ.

ε = empty string

$|X|$ = length of strings

xy = string ~~cont~~ ~~concate~~ concat length $|x| + |y|$

w is prefix of x $w \sqsubset x$ if $x = wy$ where $y \in \Sigma^*$

w is suffix of x $w \sqsupset x$ if $x = yw$ where $y \in \Sigma^*$

ε is both suffix & prefix of any string.

ab ⊢ abcca

cca ⊣ abcca

if ($m = n/2$) => $O(n^2)$

Rabin Karp Algo / Basic Idea

preprocess = $O(m)$

→ a a a a b (T) — naive Algo

→ a a b. (P)

We change our text in numeric value.

e.g a = 97 (ascii value).

So assume

a = 1
b = 2
c = 3
d = 4
e = 5
f = 6
g = 7
h = 8
i = 9
j = 10

So a a b

m = 1 + 1 + 2 = 4

hash code

function to get this value is hash function

|M| = 3

Now

N = a a a a a b
1 1 1 1 1 2

So false character = |M|

not matching, slide

a a a a a b

not matching again slide

a a a a a b

slide = rolling hash function

~~a a a~~

a a a a a b

as 4 = 4

match 4 = 4 & aab = aab

So matched

How we roll.

e.g. a b c d a b c e

1 + 2 + 3

= 6.

for shifting

(6 - 1) + 4 = 9.

b c e = 10

example.

when hash matched, check actual pattern

O(n - m +) time (avg)

⇒ Now issue of approach.

⇒ Worst Case for this

c C a CC a a e d b a

3 + 3 + 1 = 7

d b a

a + 2 + 1 = 7

→ code meteh.

→ compere alphabet

Next

again 7.

→ compere alphabet

⇒ Every time hesh code is same.

⇒ Simple hash function leed to collustons (Spurious hits)

So max time is (m n)

→ So we need a strong hash function by Robin karp.

c c a c c a a e d b a

d b a

$4 \times 10^2 + 2 \times 10^1 + 1 \times 10^0$

$P[1] \times \underbrace{10^{m-1}}_{\text{(we can take) any value}} + P[2] \times 10^{m-2} + P[3] \times 10^{m-3}$

base value = no. of ch taken if we take all english alphbets = 26 = base

if all lower, upper, special = 127

So value of P = 421.

⇒ Now we need match

c c a = 331

So not a suporious hit.

Now rolling function? How

$[(3 \times 10^2 + 3 \times 10^1 + 1 \times 10^0) - 3 \times 10^2] \times 10 + 3 \times 10^0$

⇒ Robin Finger Print function.

where base = inpute considered.

$O(n-m+1)$.

This reduces the chances of spurious hit, but still might cause it,

So worst (mn)

If we want to avoid overflow (due to big values), we will mod these values.

So what should be mod value

depends on data type if int, $\% 2^{31}$

could be any value.

↳ will get supriou hits

{ Total extended ASCII = 256 }

Algo

n → length T

m → length P

q = prime no.

h → $d^{m-1} \bmod q$ → Any random prime no.
↳ usually 256 => (here)

p = 0

$t_0 = 0$

for i = 1 to m // preprocessing

$p \rightarrow [dp + P[i]] \bmod q$

$t_0 \rightarrow [dt_0 + T[i]] \bmod q$

for S = 0 to n − m

if $p = T_0$

then

check $P[1 \text{ to } m] = T(S+1 \text{ to } S+m)$

print pattern

if $S < n - m$

$t_{S+1} = (d(t_S - T[S+1]h) + T[S+m+1]) \bmod q$

if ($t < 0$)

$t = (t + q)$