String Algorithms

size of text >> size of pattern

chk → P

Naive pattern matching

text = aafgaccaadaa bdc

pat = aabc

count = 1

$\top[\ldots n]$

$\top[\ldots m]$

$m << n$

Time Complexity:

① best case

$t \rightarrow$ aabccaadaa

$p \rightarrow$ aabc

$TC \rightarrow O(m)$

m = length of text

② worst case

→ when all chars of pat & text are same

$t \rightarrow$ aaaaaaaaa

$p \rightarrow$ aaa

$TC \rightarrow O((m-m)*m))$

$\downarrow$

$O(m*m-m)$

$t \rightarrow$ aabcabcaba

$p \rightarrow$ abcaa

Space Complexity : $O(1)$

→ Instead of matching the whole chars of pattern to the text, compare the hash values.

Hash
abc
Hash
abc
Hash
abc