String Matching Algorithms

- String Matching Algorithm is also called "String Searching Algorithm", "Pattern Searching algorithms", "Pattern Matching algorithms".
- Pattern Searching algorithms are used to find a pattern or substring from another bigger string.
- There are different algorithms.
- Given a text and pattern, we have to find all occurrences of a pattern in a text.

Problem Statement:

- Given a text array, T [1.....n], of n character (i.e. length n) and a pattern array, P [1......m], of m characters (i.e. length m) such that m <= n.
- The problem is to find an integer s, called **valid shift** where $0 \leq s < n-m$ and T [s+1......s+m] = P [1......m].
- If P occurs with shift s in T, then we say that s is a valid shift. Otherwise, we call s an invalid shift.
- We say that pattern P occurs with shift s in text T if $0 \leq s \leq n-m$ and T [s+1......s+m] = P [1......m].
- String matching problem is to find all valid shifts.
- The item of P and T are character drawn from some finite alphabet such as {0, 1} or {A, B .....Z, a, b..... z}.

Some of the algorithms used for String Matching:

1. Naive String-Matching Algorithm
2. Rabin-Karp Algorithm
3. Knuth-Morris-Pratt Algorithm (KMP algorithm)

Naive String-Matching Algorithm

- In this algorithm, we try shift s = 0, 1.......n-m, successively and for each shift s, compare T [s+1.......s+m] to P [1......m]. The naïve algorithm finds all valid shifts

Algorithm:

```
NAIVE-STRING-MATCHER (T, P)
1. n ← length [T]
2. m ← length [P]
3. for s ← 0 to n - m
4.     do if P [1.....m] = T [s + 1....s + m]
5.         then print "Pattern occurs with shift" s
```

Analysis:

For loop in line 3 executes for n-m+1 times. Inner loop at line 4 runs for m times to compare each character of P with T. So, in each iteration of s loop, we are doing m comparisons using inner loop. So, the total complexity is O((n-m+1)m).

Example:1

Text T = $\overset{1}{a}\,\overset{2}{c}\,\overset{3}{a}\,\overset{4}{a}\,\overset{5}{b}\,\overset{6}{c}$ T[1..6]

pattern P = $\overset{1}{a}\,\overset{2}{a}\,\overset{3}{b}$ P[1..3]

n = length(T) = 6 m = length(P) = 3

Shift starts from 0 to n-m

∴ 0 to 6-3

0 to 3

∴ No. of Comparisons = n-m+1 = 6-3+1 = 4

↗ No. of shifts

↱ size of outer loop.

T[1..n]	1	2	3	4	5	6	Check if P[1...m] = T[s+1...s+m]
Text (T)	a	c	a	a	b	c	
Shift 0	a	a	b				P[1..3] ≠ T[1...3]
1		a	a	b			P[1..3] ≠ T[2..4]
✓ ②			a	a	b		P[1..3] = T[3..5] (valid shift)
3				a	a	b	P[1..3] ≠ T[4..6]

Solution:

pattern occurs at shift $\boxed{s = 2}$

Example:2

$\quad\quad\quad\quad\quad 1\,2\,3\,4$

Pattern $P = 0001$ $\quad P[1..4]$

$\quad\quad\quad\quad\quad 1\,2\,3\,4\,5\,6\,7\,8\,9\,10\,11\,12\,13\,14\,15$

Text $T = 000010001010001$ $\quad T[1..15]$

$n = 15 \quad m = 4$ shift starts from 0 to $n-m$

$\Rightarrow 0$ to $15-4$

$\therefore 0$ to 11

No. of shifts $= n-m+1 = 15-4+1 = 12$

↓

No. of comparisons

$\boxed{\text{Check if } P[1..m] = T[s+1 \cdots s+m]}$

T[1⋯n]	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Text (T)	0	0	0	0	1	0	0	0	1	0	1	0	0	0	1
shift 0	0	0	0	1											
✓ ①		0	0	0	1										
2			0	0	0	1									
3				0	0	0	1								
4					0	0	0	1							
✓ ⑤						0	0	0	1						
6							0	0	0	1					
7								0	0	0	1				
8									0	0	0	1			
9										0	0	0	1		
10											0	0	0	1	
✓ ⑪												0	0	0	1

$\therefore$ pattern occurs at shift $\boxed{S = 1, 5, 11}$