

Knuth-Morris-Pratt (KMP) Pattern Matching Algorithm

- KMP algorithm is used for pattern matching. Its matching time is $O(n)$. It is achieved by avoiding backtracking on Text T.

Naïve approach – Backtracking on Text T takes place.

Example:

Text T = ^{1 2 3 4 5 6} a c a a b c T[1..6]
 Pattern P = ^{1 2 3} a a b P[1..3]

$n = \text{length}(T) = 6$ $m = \text{length}(P) = 3$
 Shift starts from 0 to $n-m$
 $\therefore 0$ to $6-3$
 0 to 3
 $\therefore$ 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
Text (T)	a	c	a	a	b	c
Shift 0	a	a	b			
1		a	a	b		
2			a	a	b	
3				a	a	b

check if $P[1..m] = T[s+1..s+m]$
 $P[1..3] \neq T[1..3]$
 $P[1..3] \neq T[2..4]$
 $P[1..3] = T[3..5]$ (valid shift)
 $P[1..3] \neq T[4..6]$

Solution:
 pattern occurs at shift $s = 2$

- Characters in the text are repeatedly compared in Naïve algorithm (basic string-matching algorithm). Whenever, there is a mismatch backtracking happens on Text.
- Reason for backtracking is Naïve approach does not study the pattern. Hence, backtracks whenever there is a mismatch.
- This backtracking is avoided in KMP algorithm.

KMP studies the pattern. KMP algorithm has 2 components:

1. Prefix function – Phi value computation (prepare Phi table (or) Prefix table)
2. KMP matcher – performs pattern matching

Algorithm:

KMP-MATCHER(T, P)

```
1   $n \leftarrow \text{length}[T]$ 
2   $m \leftarrow \text{length}[P]$ 
3   $\pi \leftarrow \text{COMPUTE-PREFIX-FUNCTION}(P)$ 
4   $q \leftarrow 0$  ▷ Number of characters matched.
5  for  $i \leftarrow 1$  to  $n$  ▷ Scan the text from left to right.
6      do while  $q > 0$  and  $P[q + 1] \neq T[i]$ 
7          do  $q \leftarrow \pi[q]$  ▷ Next character does not match.
8      if  $P[q + 1] = T[i]$ 
9          then  $q \leftarrow q + 1$  ▷ Next character matches.
10     if  $q = m$  ▷ Is all of  $P$  matched?
11         then print “Pattern occurs with shift”  $i - m$ 
12          $q \leftarrow \pi[q]$  ▷ Look for the next match.
```

COMPUTE-PREFIX-FUNCTION(P)

```
1   $m \leftarrow \text{length}[P]$ 
2   $\pi[1] \leftarrow 0$ 
3   $k \leftarrow 0$ 
4  for  $q \leftarrow 2$  to  $m$ 
5      do while  $k > 0$  and  $P[k + 1] \neq P[q]$ 
6          do  $k \leftarrow \pi[k]$ 
7      if  $P[k + 1] = P[q]$ 
8          then  $k \leftarrow k + 1$ 
9       $\pi[q] \leftarrow k$ 
10 return  $\pi$ 
```

Step-1: Compute prefix table (Phi table) for pattern P.

Prefix function trace out for Phi table value computation

$P: a \ b \ a \ b \ a \ c \ a$ π Trace out

$m = 7$ $\pi(1) = 0$ $k = 0$

For $q = 2$ to 7 .

$q = 2$ while $k > 0$ and $P[k+1] \neq P[q]$ π $d = p$
 $k = 0$ $k = \pi(k)$ π $d = p$
 $0 > 0 - F$ π $d = p$
 If $P[k+1] = P[q]$ $k = k + 1$ π $d = p$
 $P[1] = P[2]$ π $d = p$
 $a = b$ $-F$ π $d = p$

	$\pi(q)$	k
	$\pi(2) = 0$	
P	1	2
	a	b
π	0	0

$q = 3$ while $(k > 0) - F$
 $k = 0$ $0 > 0$
 If $P[k+1] = P[q] \Rightarrow P[1] = P[3] - T$
 $k = k + 1 = 1$
 $\pi[2] = k$
 $\pi[3] = 1$

	1	2	3	4	5	6	7
P	a	b	a	b	a	c	a
π	0	0	1				

$q = 4$ while $(k > 0) - T$ and $P[k+1] \neq P[q]$
 $P[2] \neq P[4] \Rightarrow b \neq b - F$
 $k = 1$ If $P[k+1] = P[q]$
 $P[2] = P[4] \Rightarrow b = b - T$
 $k = k+1 = 2$
 $\pi[q] = k$
 $\pi[4] = 2.$

	1	2	3	4	5	6	7
P	a	b	a	b	a	c	a
π	0	0	1	2	3	0	1

$q = 5$
 $k = 2$

	1	2	3	4	5	6	7
P	a	b	a	b	a	c	a
π	0	0	1	2	3	0	1

$q = 6$
 $k = 3$

	1	2	3	4	5	6	7
P	a	b	a	b	a	c	a
π	0	0	1	2	3	0	1

$q = 7$
 $k = 0$

	1	2	3	4	5	6	7
P	a	b	a	b	a	c	a
π	0	0	1	2	3	0	1

$\therefore$ prefix table (π) for pattern P

	1	2	3	4	5	6	7
P	a	b	a	b	a	c	a
π	0	0	1	2	3	0	1

Practice: Compute Phi value for the following patterns

P1: a b a b d

P2: a b c d a b e a b f

P3: a b c d e a b f a b c

Solution:

P1: a b a b d

1	2	3	4	5
a	b	a	b	d
0	0	1	2	0

P2: a b c d a b e a b f

1	2	3	4	5	6	7	8	9	10
a	b	c	d	a	b	e	a	b	f
0	0	0	0	1	2	0	1	2	0

P3: a b c d e a b f a b c

1	2	3	4	5	6	7	8	9	10	11
a	b	c	d	e	a	b	f	a	b	c
0	0	0	0	0	1	2	0	1	2	3

KMP Matcher Traceout

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
T	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

P	1	2	3	4	5	6	7
	a	b	a	b	a	c	a
Phi	0	0	1	2	3	0	1

$n = 15, m = 7, q=0$

for loop from $i = 1$ to $n = 1$ to 15

In each iteration we compare $T[i]$ with $P[q+1]$

Iteration-1: $i = 1$

	i														
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

	q							
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare $T[i]$ with $P[q+1]$ – b with a - not match

As q is at 0, we cannot move q. So, move i to next position (i.e) next iteration

Iteration-2: i = 2

		i													
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

	q							
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – a with a - match

Move both q (i.e) q = q+1 and i to next position

Iteration-3: i = 3

			i												
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

	q							
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – c with b – not match

Move q to the corresponding Phi value b'coz q > 0.

q is at 1. Corresponding Phi value is 0. So, q = 0.

As q is at 0, further we cannot move q. so move i to next position.

Iteration-4: i = 4

				i											
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

	q							
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – b with a – not match

As q is at 0, further we cannot move q. So, move i to next position.

Iteration-5: i = 5

					i										
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

	q							
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – a with a – match

Move i and q to next position

Iteration-6: i = 6

						i									
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

		q						
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – b with b – match

Move i and q to next position

Iteration-7: i = 7

							i								
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

			q					
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – a with a – match

Move i and q to next position

Iteration-8: i = 8

								i							
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

				q				
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – b with b – match

Move i and q to next position

Iteration-9: i = 9

									i						
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

					q			
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – a with a – match

Move i and q to next position

Iteration-10: i = 10

										i					
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

						q		
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – b with c – not match

Move q to the corresponding Phi value.

q is at 5. Corresponding Phi value is 3. So, q = 3.

				q				
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Continue comparison

Compare T[i] with P[q+1] – b with b – match

Move i and q to next position

Iteration-11: i = 11

											i				
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

P					q			
	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – a with a – match

Move i and q to next position

Iteration-12: i = 12

T												i			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

P						q		
	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – c with c – match

Move i and q to next position

Iteration-13: i = 13

T													i		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

							q	
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare $T[i]$ with $P[q+1]$ – a with a – match

Move q to next position i.e $q = 7$

In this iteration, q becomes equal to m (i.e.) 7. So, pattern has occurred in the text T. Pattern occurs with shift $i - m = 13 - 7 = 6$.

Print the i-m value as valid shift number.

Note: In each iteration, we check whether $q = m$. If $q = m$ condition is true, then print that the pattern occurred at shift $i - m$. Once you print, immediately move q to its corresponding Phi value. Here corresponding Phi value is 1, so we move q to 1.

Then move i to next position

Iteration-14: i = 14

Here, $q = 1$

														i	
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

		q						
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – c with b – not match

Move q to the corresponding Phi value.

q is at 1. Corresponding Phi value is 0. So, q = 0.

As q is at 0, further we cannot move q. So, move i to next position.

Iteration-15: i = 15

															i
T	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	b	a	c	b	a	b	a	b	a	b	a	c	a	c	a

	q							
P	0	1	2	3	4	5	6	7
		a	b	a	b	a	c	a
		0	0	1	2	3	0	1

Compare T[i] with P[q+1] – a with a – match

Move i and q to next position (i.e) next iteration

With this loop terminates as i = 16 exceeds the n value (n = 15). Algorithm terminates