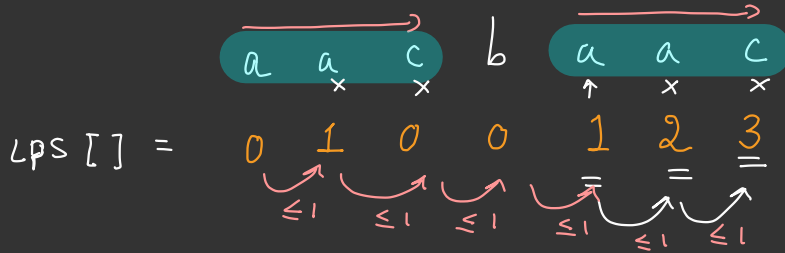


"Pattern Matching"

Pattern \rightarrow abc

Text \rightarrow xyz abc at abc

Concept . LPS



$O(N^3)$ Algo (Brute Force)

$\rightarrow O(N)$ Algo (Pseudo Code) $\leftarrow ?$

"Hardest Algo in CS"

KMP Algorithm

Scientists { Knuth
Morris
Pratt } 1970-77

Calc LPS[]



4

5

$LPS[i]$
 $= 5$

Assumption:

all values till

$LPS[i-1]$ are
known

Observation:

$$\underbrace{S_0 S_1 S_2 S_3 S_4}_{\text{crossed out}} = \underbrace{S_{i-4} S_{i-3} S_{i-2} S_{i-1} S_i}_{\text{crossed out}}$$

Claim:

Observation-1

$LPS[i]$ exceeds $LPS[i-1]$ by at most 1

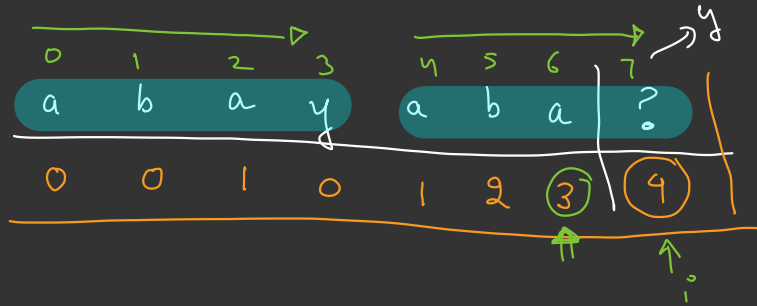
$S =$
 $LPS =$

$c \ a \ c \ y \ c \ a \ c \ a \ z$
 $0 \ 0 \ 1 \ 0 \ 1 \ 2 \ 3 \ 2 \ 0$

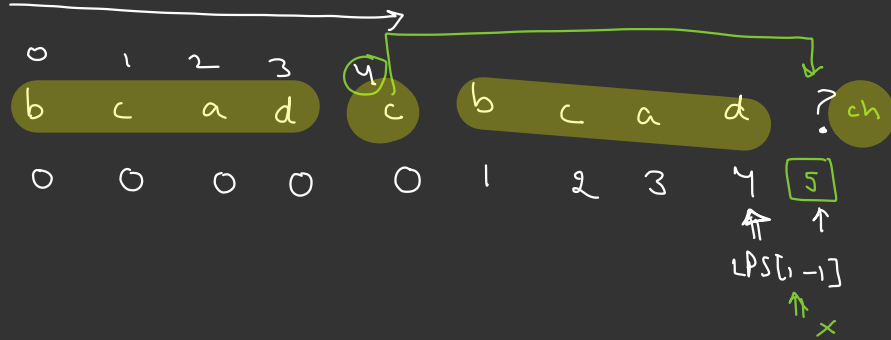
inc
 can never
 be greater
 than 1

At max
claim $LPS[i] = LPS[i-1] + 1$

Step - 2

$$\frac{3x-1}{-}$$
$$S =$$
$$\text{LPS}[] =$$

$$X = \text{LPS}[i-1]$$
$$X \downarrow$$
$$S[i] = S[3]$$
$$\underline{\epsilon_{x-2}}$$

LPS =

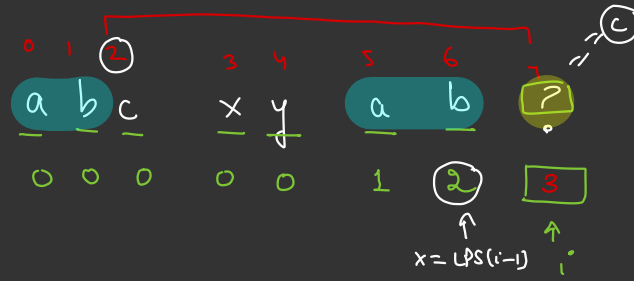

$$c_h = c$$
$$x = \text{LPS}[1-1]$$

```
if (s[x] == s[i]) {
```

$$LPS[i] = LPS[i-1] + 1$$

3

Ex-3



$$x = \text{LPS}(i-1)$$

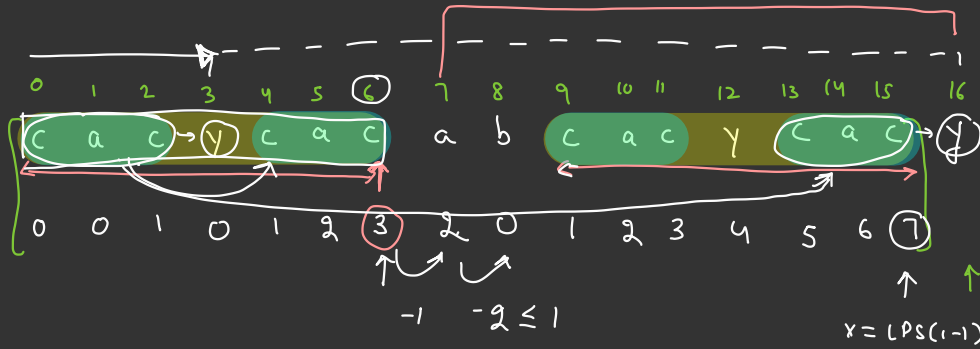
$$\text{if } (s[i] == s[x])$$

$$\text{LPS}(i) = 1 + \text{LPS}(i-1)$$

Step -3

$E(x -)$

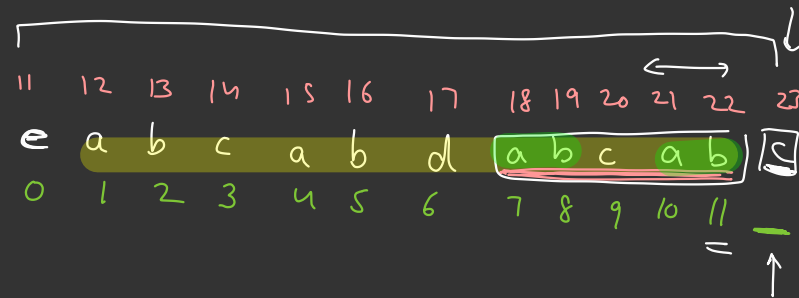
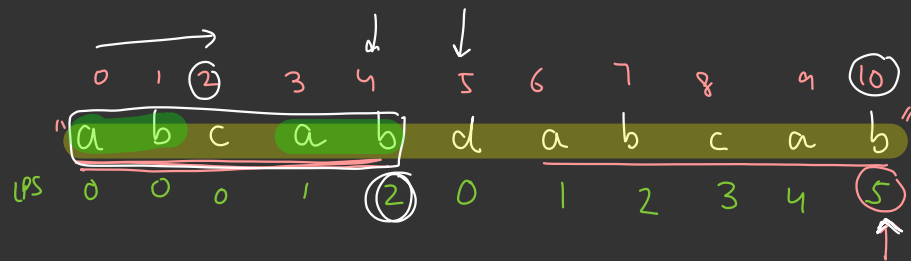
LPS []



$j = 16$

x	str[i] == str[x]	
1	$s[16] == s[1]$ $y == a$	No $x = \text{LPS}[x-1]$ $=$ $x = \text{LPS}[6]$ $= 3$
3	$s[16] == s[3]$ $y == y$	Yes $\text{LPS}[i] = x + 1$ $= 3 + 1$ $= 4$

finding
subpatterns
that can
possibly
match



$$x = \text{LPS}[i-1]$$

$$= 11$$

Loop	x	str[x] == str[i]		
	11	e == c	No	$x = \text{LPS}[x-1]$
	5	d == c	No	$x = \text{LPS}[x-1]$
	2	c == c	Yes	$\text{LPS}[i] = \underline{x+1}$ $= 3$

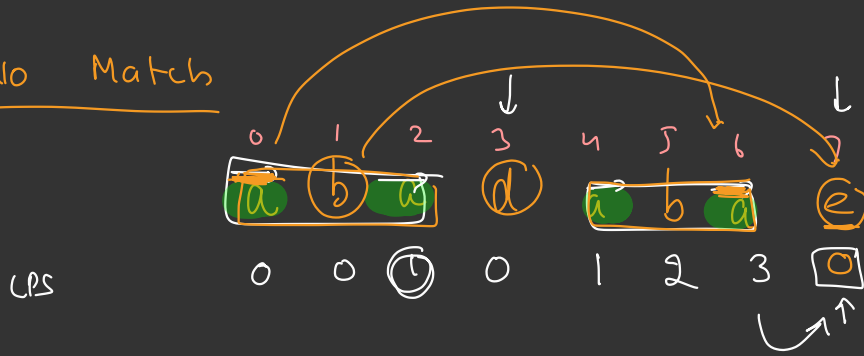
abcab[2]
↑
s

ab□

abcab[?]
↑
LPS =
s

ab□

No Match



LPS

$$x = \text{LPS}(i-1) = 3$$

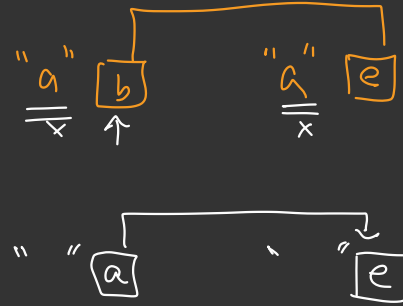
x	str[x] == str[i]	
3	d == e	(No)
1	b == e	(No)
0	a == e	(No)

$x = \text{LPS}(x-1)$ Reducing pattern length

$\rightarrow x = \text{LPS}(x-1)$

$$x = \text{LPS}(x-1) = \text{LPS}[-1] = ?$$

Sp case if $(x = 0 \text{ \& \& no match})$
break;



Pseudo Code -

Lps Array (String S)

$N = S.length()$

$LPS[N]$,

$\rightarrow LPS[0] = 0$

$\rightarrow \text{for } (i=1 \text{ to } i < N) \{$

$x = LPS[i-1]$

$\text{while } (S[x] \neq S[i]) \{$

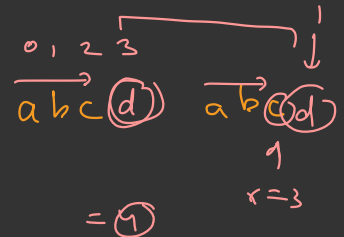
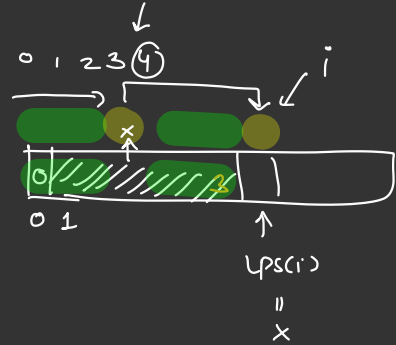
$\text{if } (x == 0) \{ \underline{x = -1}, \text{ break; } \}$

$x = LPS[x-1]$

$\}$

$\rightarrow LPS[i] = x + 1; \leftarrow \text{Match}$

$\}$



Q) Given a string of len N ,
 calc min no of chars needed
 to add at start to make entire string as



10.40

palindrome

Ex-1

+
[dc]

Left
 \longleftrightarrow
 a b a c d

Ex-2

+
[dca]

a b b a a c d
 \longleftrightarrow

longest

prefix

which is

a palindrome

$$\text{ans} = N - \text{len of longest prefix palindrome}$$

Ex-3

aba

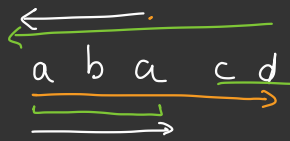
^{1 2 3 4 5 6 7}
c d d c a b a

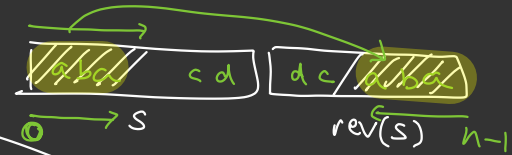
Ex-4

a f e d c b
↑
Pattern

@ b c d e f a

Logic

$s =$ 


 s $rev(s)$ $n-1$

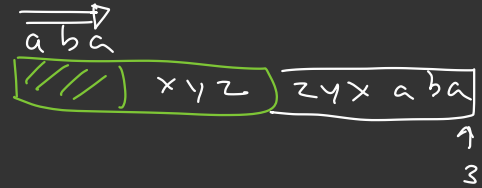
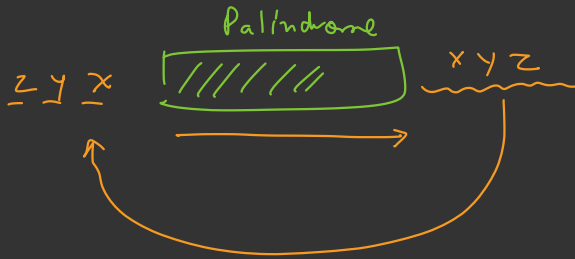
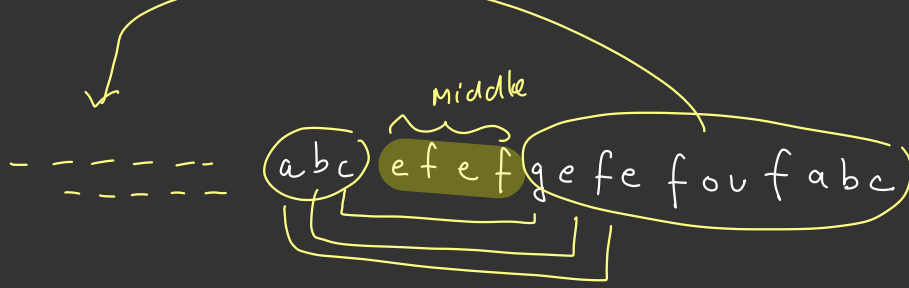
$$s' = \underline{s + rev(s)} =$$


 $LPS[] =$ 0 0 1 0 0 0 0 1 2 3

length
of longest
prefix
palindrome

$$\begin{aligned} N - \underline{LPS(\text{last_idx})} \\ = 5 - 3 \\ = 2 \end{aligned}$$

$S =$ "a a a a"



$$S + "$" + \text{rev}(S)$$

\Rightarrow `a a a a` `a a a a`
 $\text{LPS}[7] =$ `0 1 2 3 4 5 6 7`

`a` `a a a` \$ `a` `a a a`
`0 1 2 3 0 1 2 3` `4`

$$\begin{aligned}
 &N - \text{len} \\
 &= 4 - 4 \\
 &= 0
 \end{aligned}$$

$s = \underline{a b a c d}$

LPS =

a	b	a	c	d	\$	d	c	a	b	a
0	0	1	0	0	0	0	0	1	2	3

↑

$N = 5$

$\overbrace{d c \boxed{a b a} c d}^{\text{a}}$

$$\begin{aligned} \text{ans} &= 5 - \text{Lps} \\ &= 5 - 3 \\ &= \boxed{2} \end{aligned}$$

Hashing

Friday
Sunday