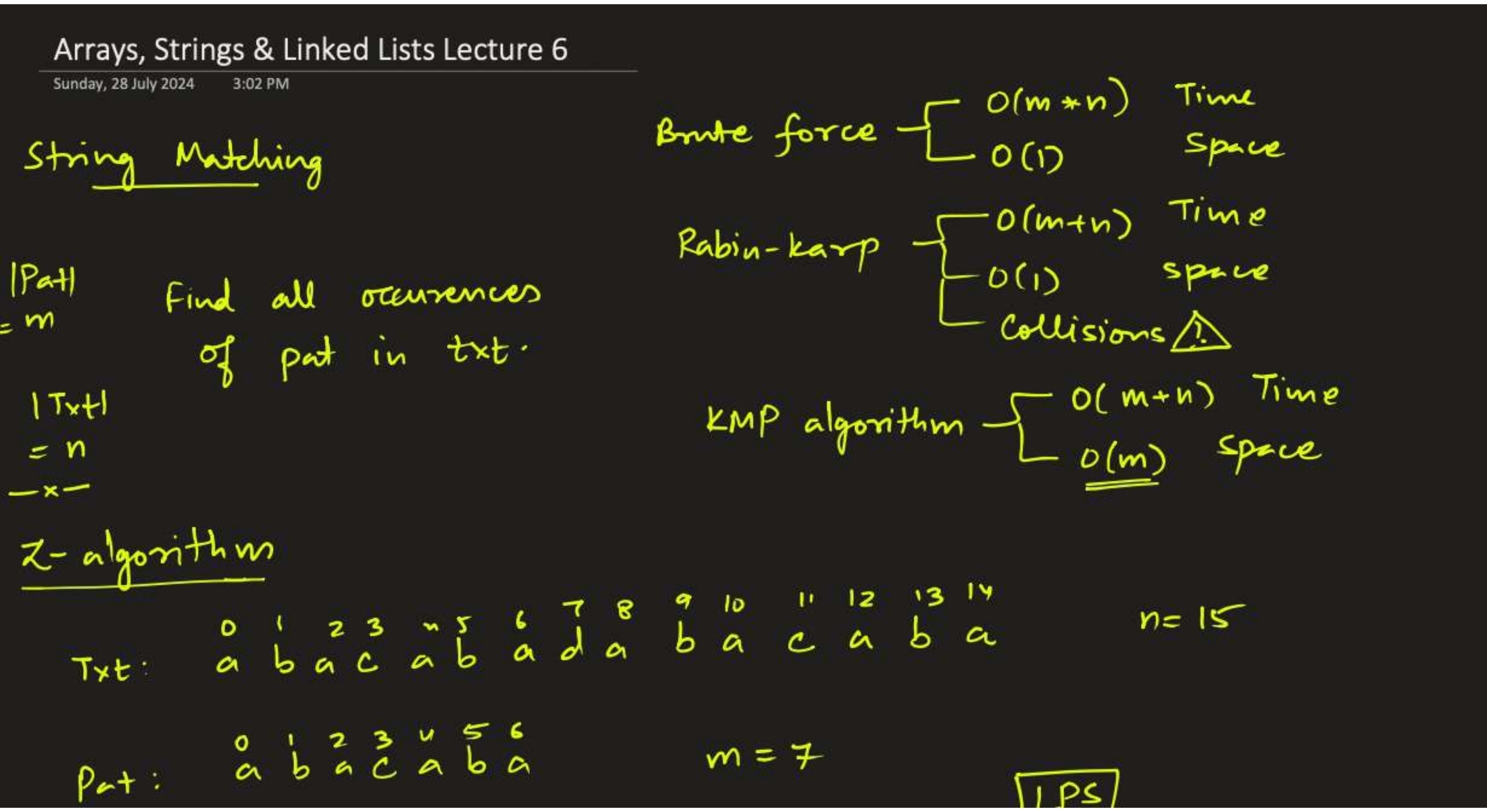


Arrays, Strings & Linked Lists Lecture 6

Sunday, 28 July 2024 3:02 PM

String Matching

|Pat| = m

|Txt| = n

Find all occurences of pat in txt.

—x—

Brute force { O(m * n) Time
O(1) Space

Rabin-karp { O(m+n) Time
O(1) space
Collisions ⚠

KMP algorithm { O(m+n) Time
$\underline{\underline{O(m)}}$ Space

Z- algorithm

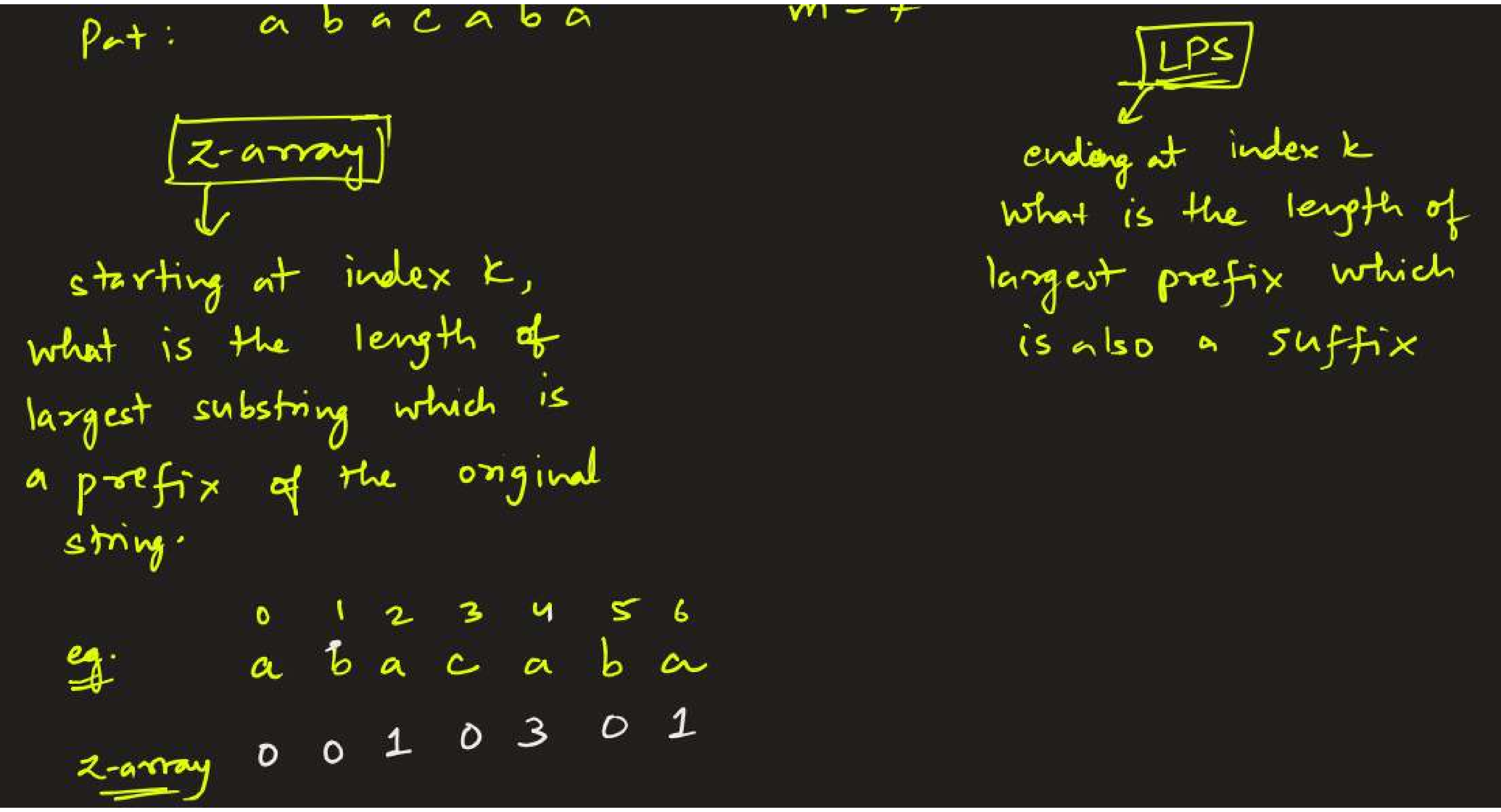
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Txt:	a	b	a	c	a	b	a	d	a	b	a	c	a	b	a

n= 15

	0	1	2	3	4	5	6
Pat:	a	b	a	c	a	b	a

m = 7

$\boxed{\text{LPS}}$



Pat : a b a c a b a   m = 7

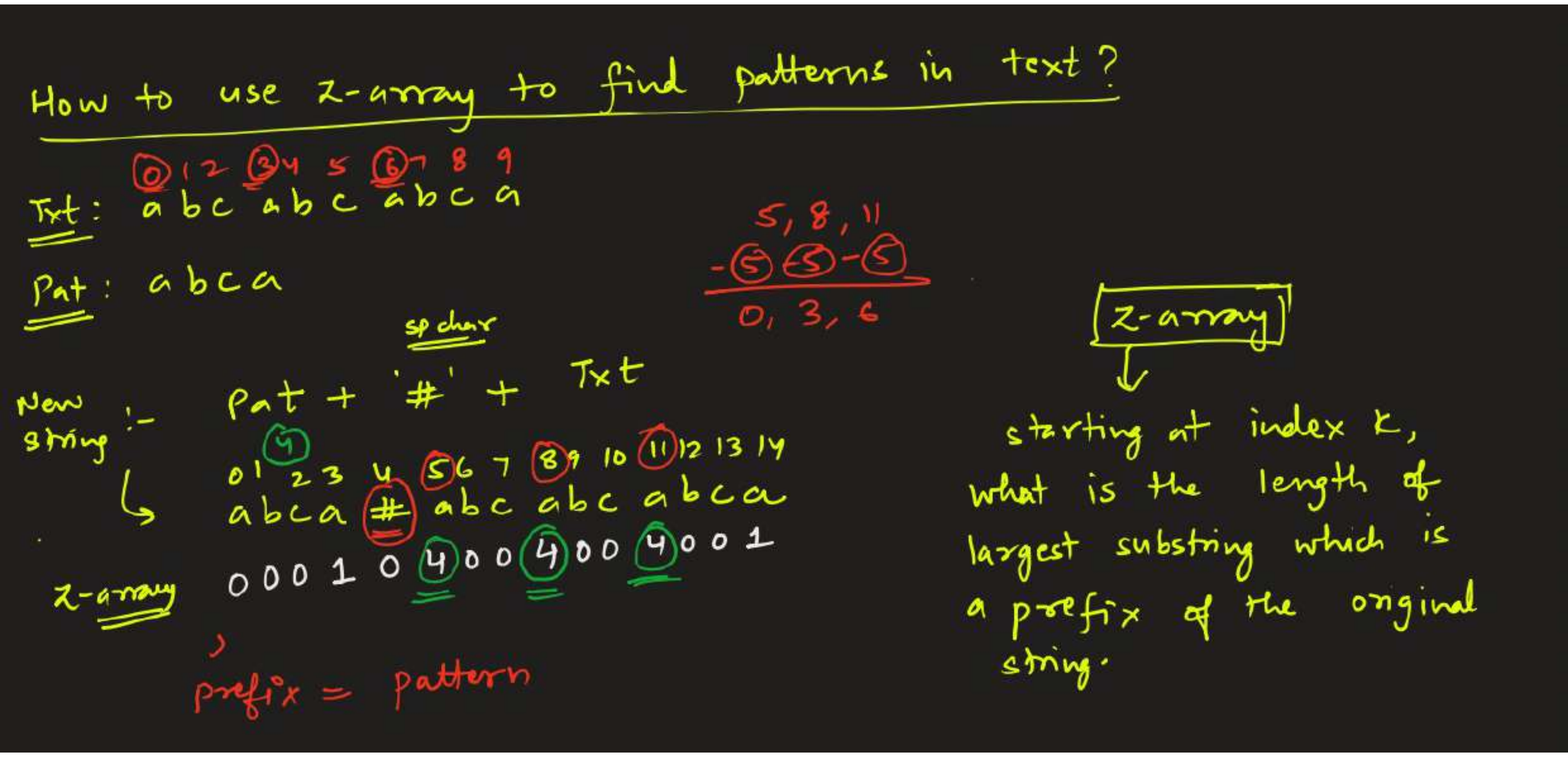
Z-array

starting at index k, what is the length of largest substring which is a prefix of the original string.

	0	1	2	3	4	5	6
eg.	a	b	a	c	a	b	a
Z-array	0	0	1	0	3	0	1

LPS

ending at index k what is the length of largest prefix which is also a suffix



How to use z-array to find patterns in text?

0 1 2 3 4 5 6 7 8 9

Txt : a b c a b c a b c a

Pat : a b c a

5, 8, 11

-5 -5 -5

0, 3, 6

sp char

New string :- Pat + '#' + Txt

4

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

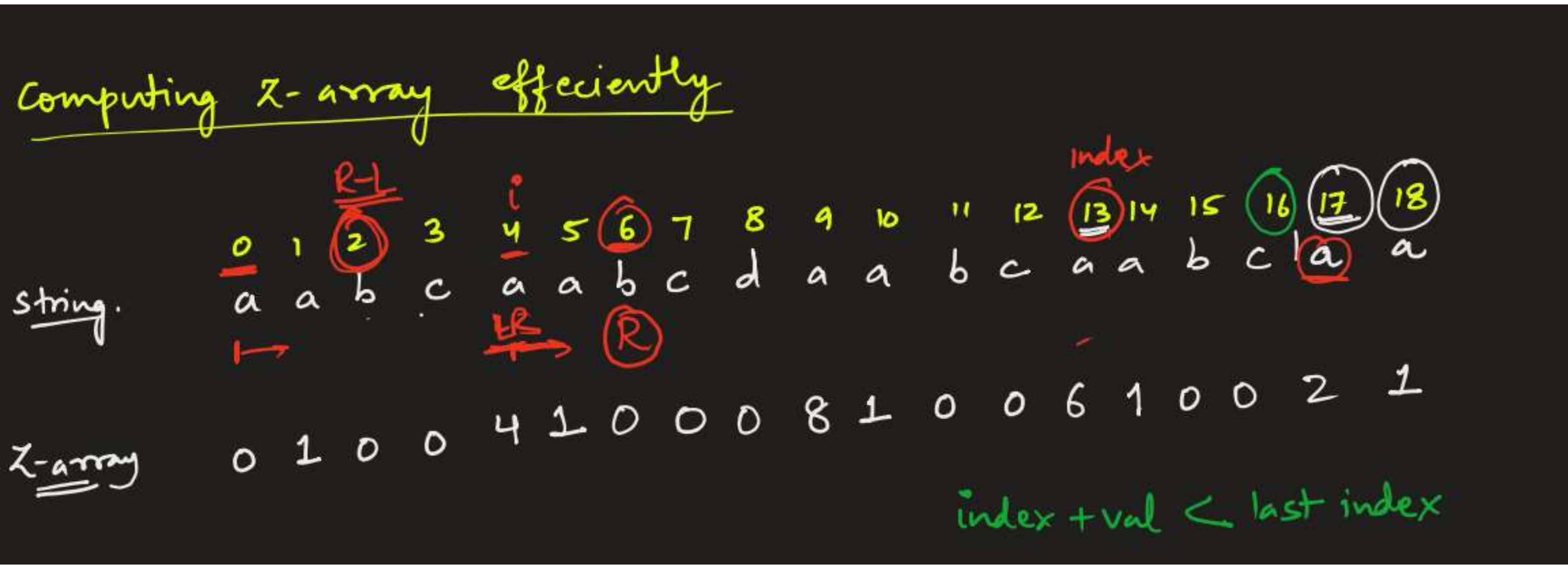
a b c a # a b c a b c a b c a

z-array 0 0 0 1 0 4 0 0 4 0 0 4 0 0 1

prefix = pattern

Z-array

starting at index k, what is the length of largest substring which is a prefix of the original string.



Computing Z-array effeciently

R-L

i

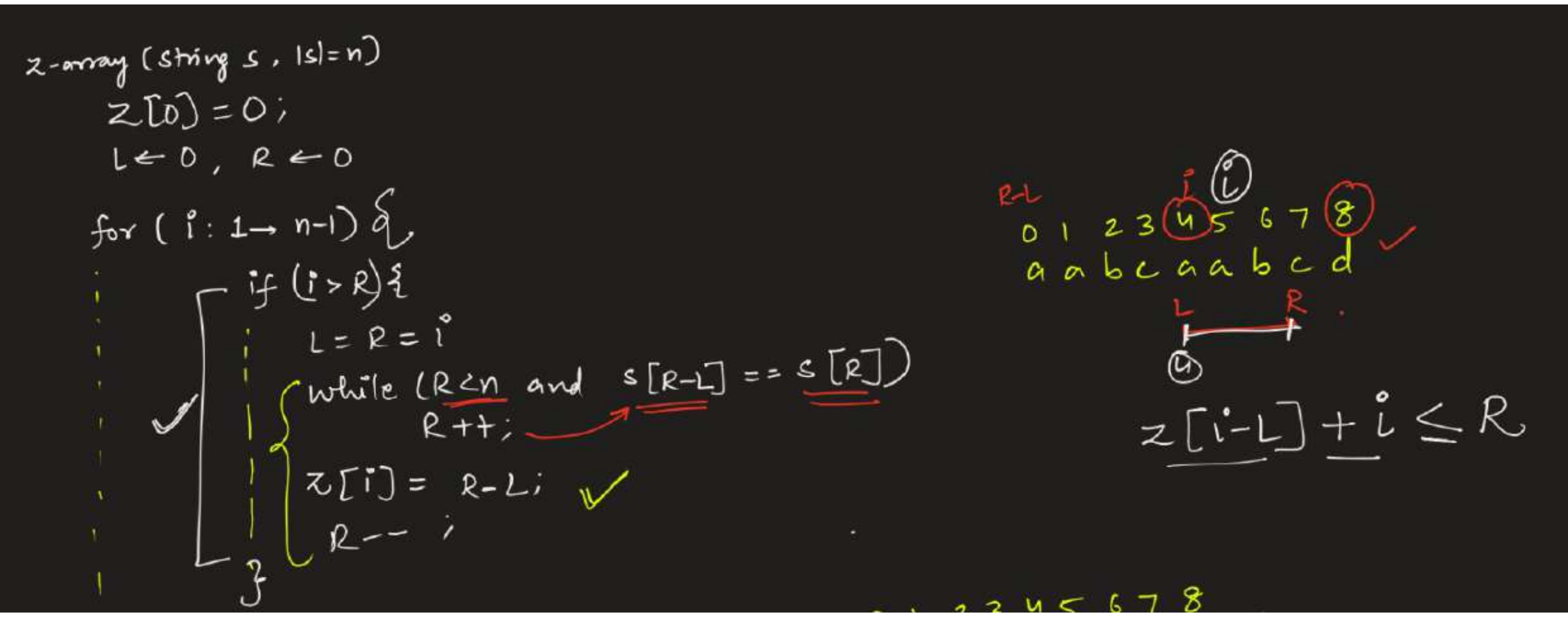
index

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
string.	a	a	b	c	a	a	b	c	d	a	a	b	c	a	a	b	c	a	a
Z-array	0	1	0	0	4	1	0	0	0	8	1	0	0	6	1	0	0	2	1

LR

R

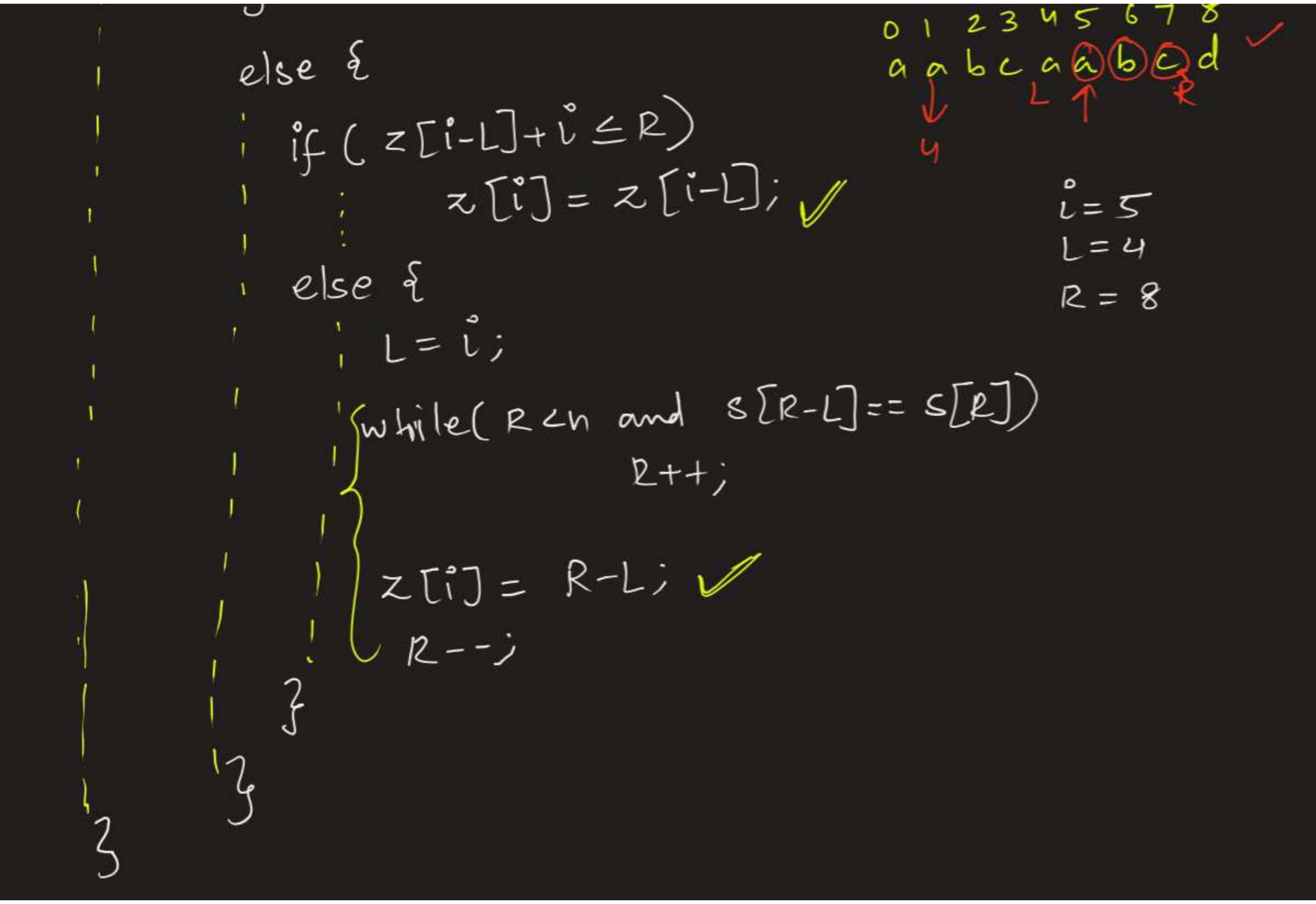
index + val < last index



```
z-array (string s , |s|=n)
    Z[0] = 0;
    L ← 0 , R ← 0

    for ( i : 1 → n-1) {
        if (i > R) {
            L = R = i
            while (R<n and s[R-L] == s[R])
                    R++;
            z[i] = R-L;
            R-- ;
        }
```

$z[i-L] + i \le R$



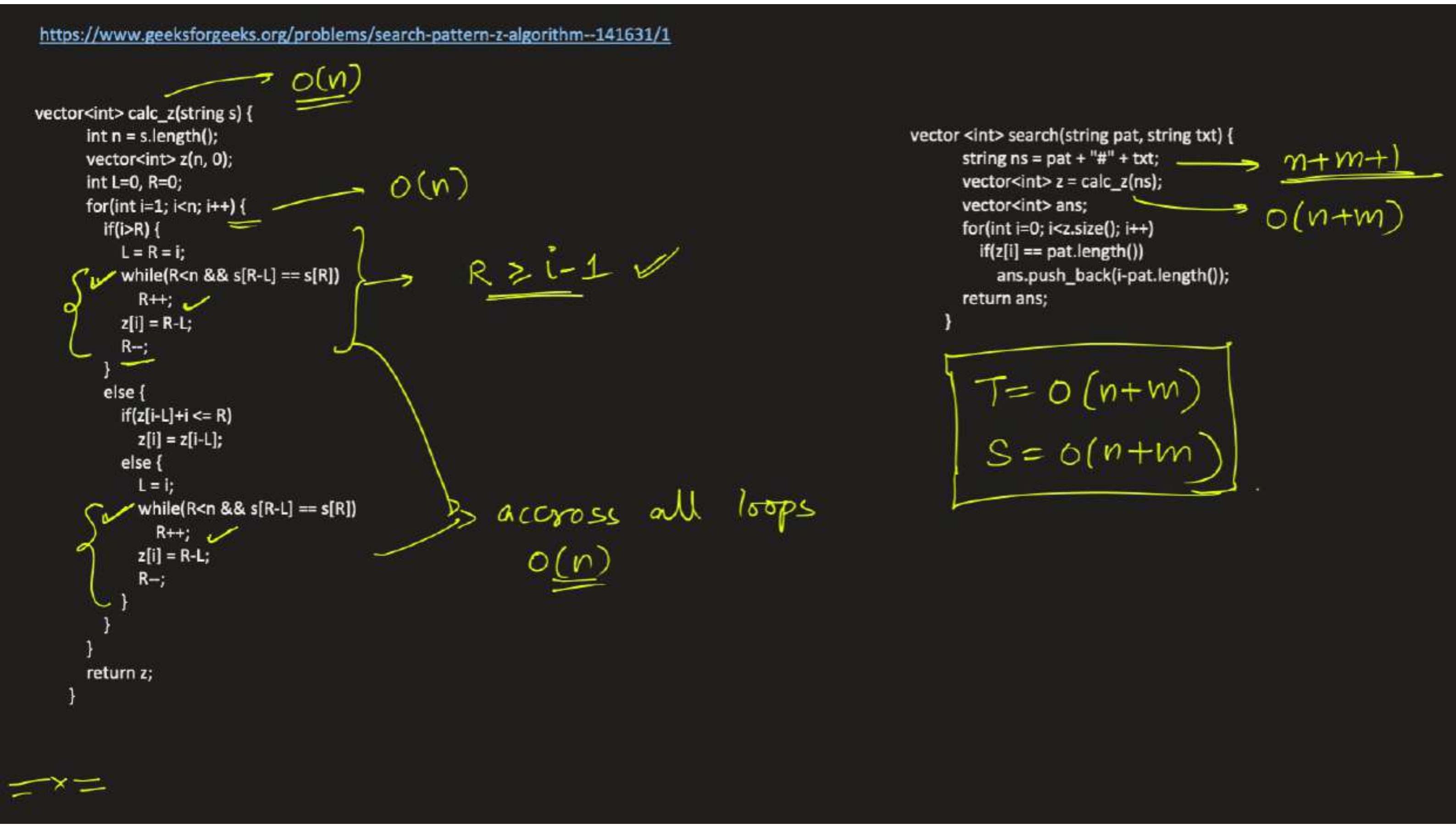
```
else {
    if ( z[i-L]+i ≤ R)
           z[i] = z[i-L]; ✓

    else {
        L = i;
        while( R<n and s[R-L]== s[R])
                   R++;

        z[i] = R-L; ✓
        R--;
    }
}
}
```

```
0 1 2 3 4 5 6 7 8
a a b c a a b c d ✓
  ↓       L ↑   R
  4
```

i = 5
L = 4
R = 8



https://www.geeksforgeeks.org/problems/search-pattern-z-algorithm--141631/1

```
vector<int> calc_z(string s) {
        int n = s.length();
        vector<int> z(n, 0);
        int L=0, R=0;
        for(int i=1; i<n; i++) {
           if(i>R) {
              L = R = i;
              while(R<n && s[R-L] == s[R])
                 R++;
              z[i] = R-L;
              R--;
           }
           else {
              if(z[i-L]+i <= R)
                 z[i] = z[i-L];
              else {
                 L = i;
                 while(R<n && s[R-L] == s[R])
                    R++;
                 z[i] = R-L;
                 R--;
              }
           }
        }
        return z;
     }
```

$O(n)$

$O(n)$

$R \geq i-1$ ✓

accross all loops $O(n)$

```
vector <int> search(string pat, string txt) {
        string ns = pat + "#" + txt;
        vector<int> z = calc_z(ns);
        vector<int> ans;
        for(int i=0; i<z.size(); i++)
           if(z[i] == pat.length())
              ans.push_back(i-pat.length());
        return ans;
     }
```

$n+m+1$

$O(n+m)$

$$\boxed{\begin{aligned} T &= O(n+m) \\ S &= O(n+m) \end{aligned}}$$

=x=