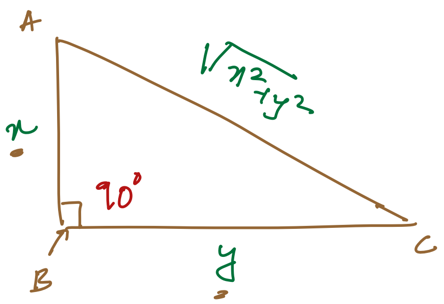
Today's Content:

- Given N points No: of triangles can be formed
- Given N points No: of rectangles can be formed
- {1 more problem}

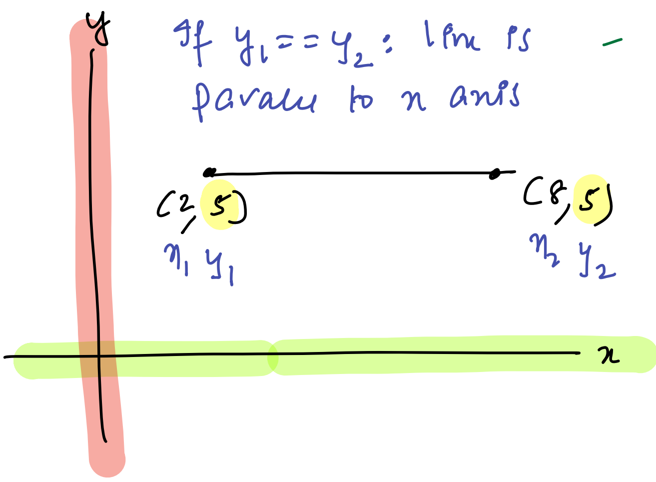
1Q) Right Angled Triangle Basics → {10:25 PM}



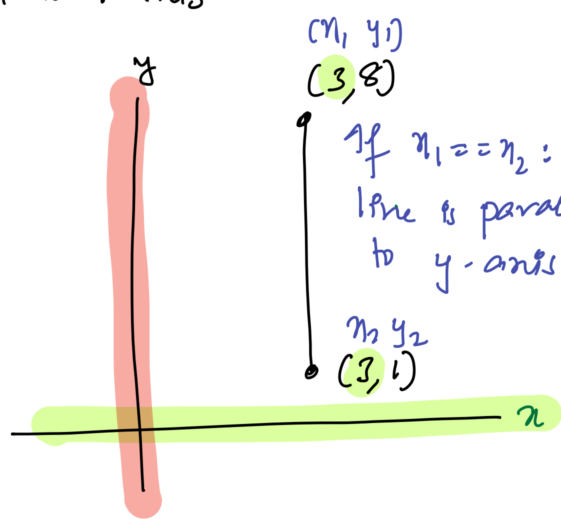
AC – (longest side) : Hypotenuse

2Q) Parallel to X-axis / Parallel to Y-axis

If $y_1 == y_2$: line is parallel to x axis

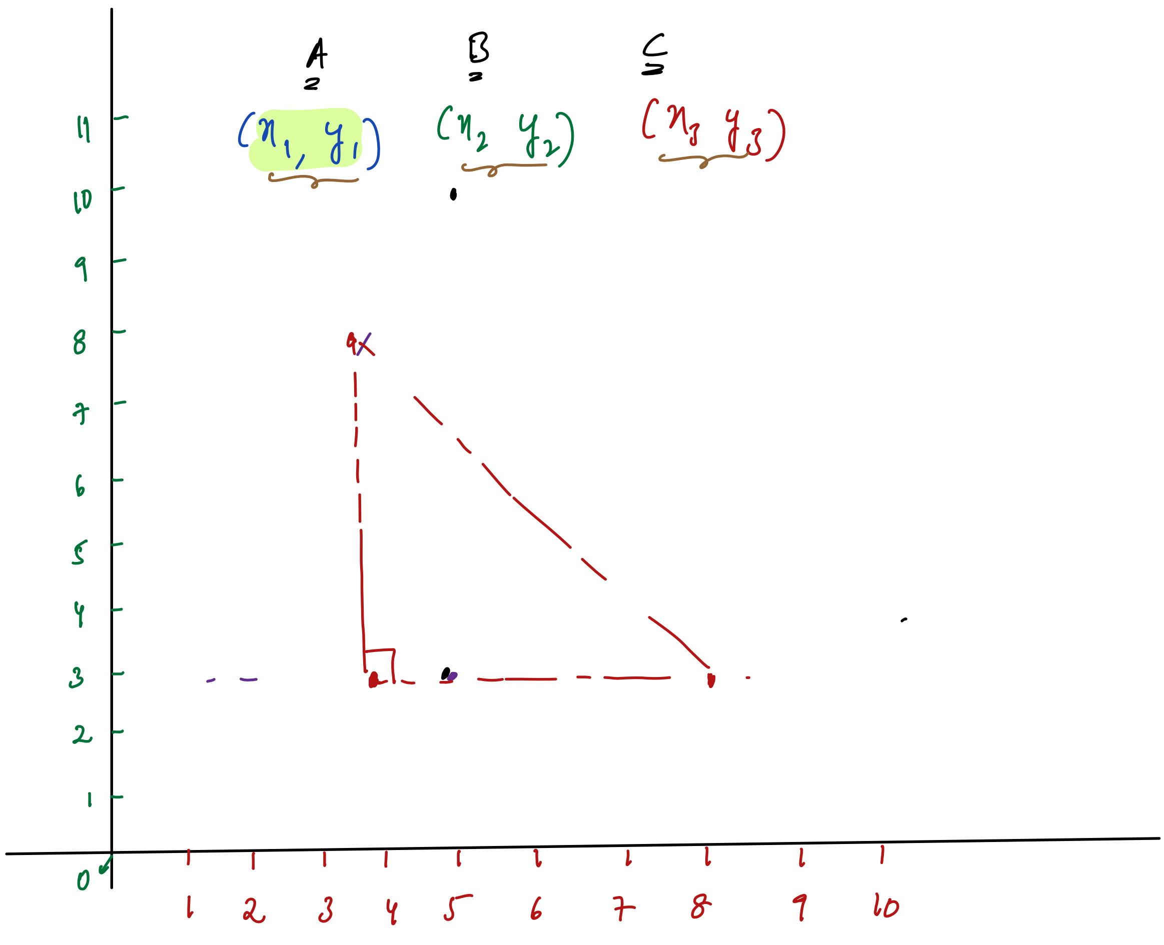


If $x_1 == x_2$: line is parallel to y-axis

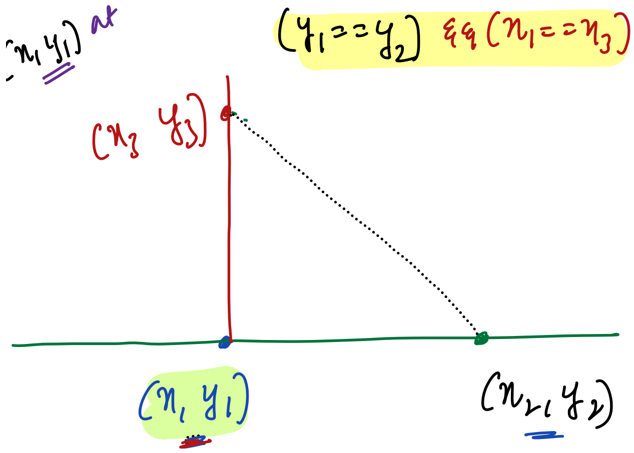


28) Given 3 distinct points in a 2D plane check if they form triangle, such that shorter sides are parallel to x-axis & y-axis respectively

	A	B	C
Ex1:	(1, 8)	(1, 4)	(5, 4)
Ex2:	(5, 10)	(1, 3)	(5, 3)
Ex3:	(4, 3)	(8, 3)	(4, 8)

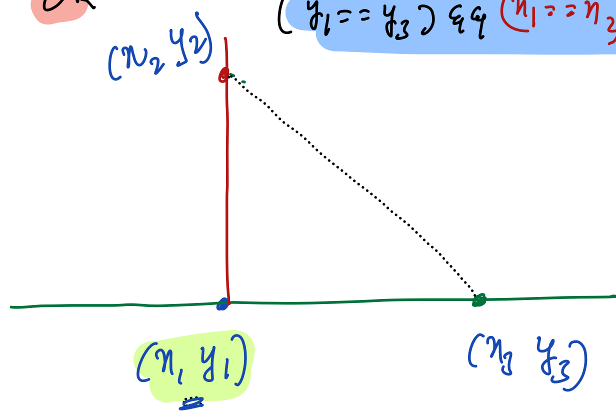


$(y_1 == y_2)$ && $(x_1 == x_3)$



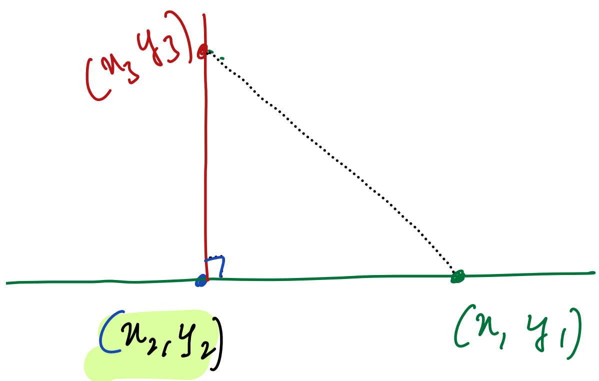
OR

$(y_1 == y_3)$ && $(x_1 == x_2)$



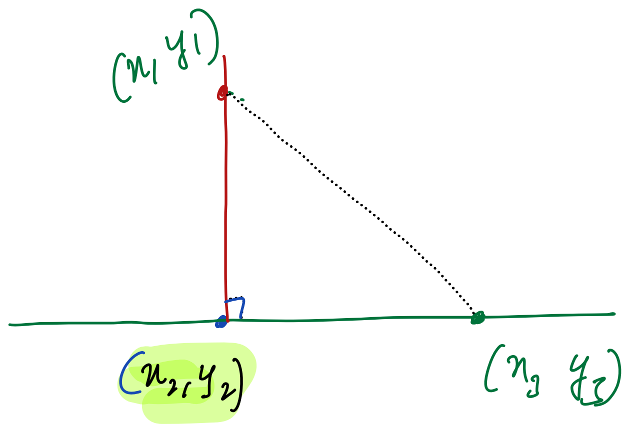
OR

$(y_2 == y_1)$ && $(x_2 == x_3)$



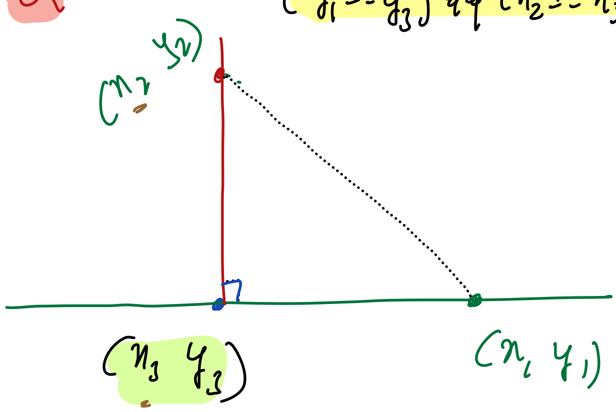
OR

$(y_2 == y_3)$ && $(x_1 == x_2)$



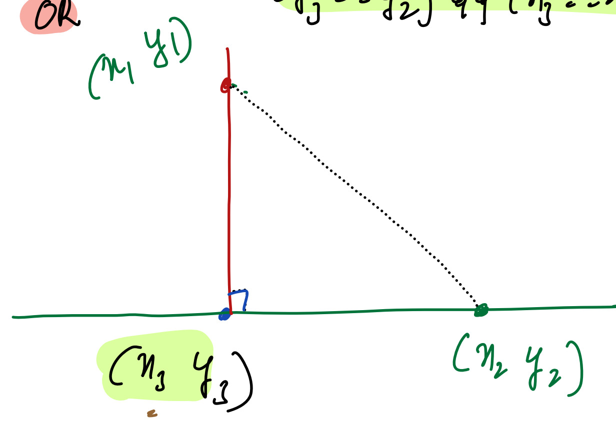
OR

$(y_1 == y_3)$ && $(x_2 == x_3)$



OR

$(y_3 == y_2)$ && $(x_3 == x_1)$



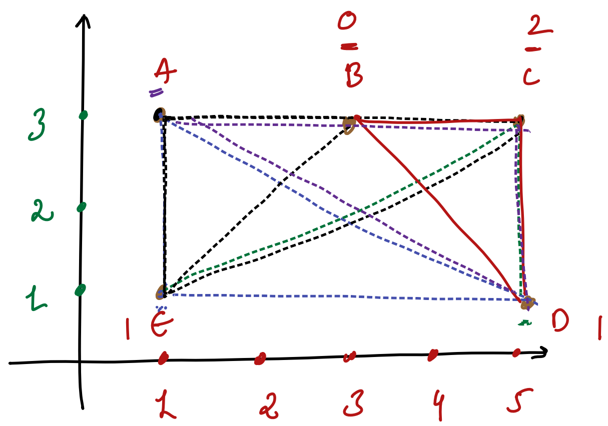
// given 3 points (x_1, y_1) $(x_2\ y_2)$ $(x_3\ y_3)$ → ◺, || to x-axis || to y-axis

→ TODO: TC: $O(1)$ SC: $O(1)$

35) Given **N distinct points** in a 2D plane, calculate no. of triangles are formed such that **shorter sides** are parallel to x-axis & y-axis

Note: 2 Arrays $x[N]$, $y[N]$ are given. such that $i^{th} = (x[i], y[i])$

Ex:



N = 5 points

ans = 6

AED   EAC
CDE   DCB
EAB   DCA

BF: For every triplet, check if it forms a △, side are $\parallel$ to x & y axis

TC: $O(N^3)$ i: ______
j: ______
k: ______

SC: $O(1)$

Input:

Ex:

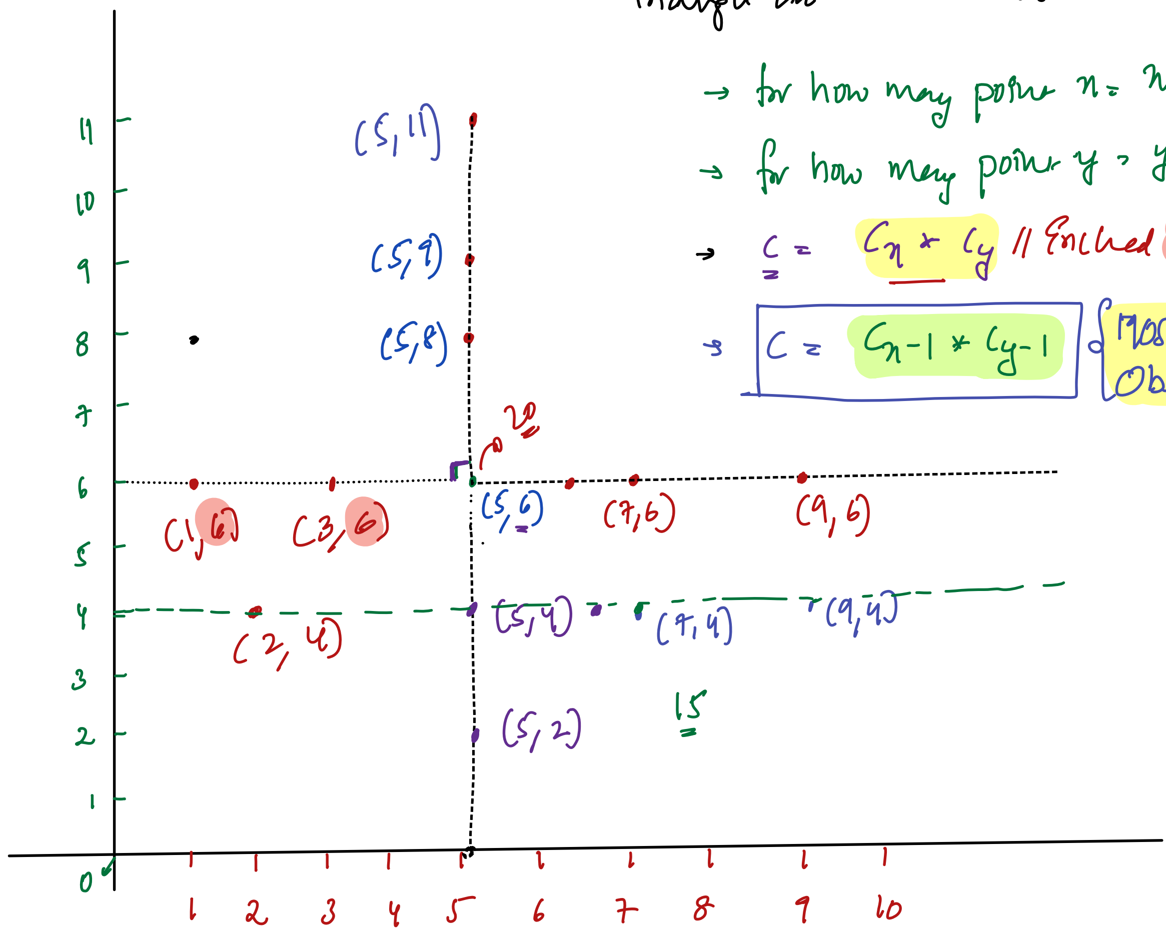
obs: (x_i, y_i) how many right angle triangle are with this as center?

→ for how many points $x = x_i = C_x$

→ for how many points $y = y_i = C_y$

→ $C = C_x * C_y$ // included x_i & y_i

→ $\boxed{C = C_x - 1 * C_y - 1}$ {MOST IMP Obs}



→ no:of points with given x

// Idea: Use 2 maps, hmx, hmy → no:of points with given y

Hashmap<int, int> hmx, hmy;

iterate in all points (insert x in hmx & y in hmy)

TC: $O(N)$

SC: $O(2N)$ → $O(N)$

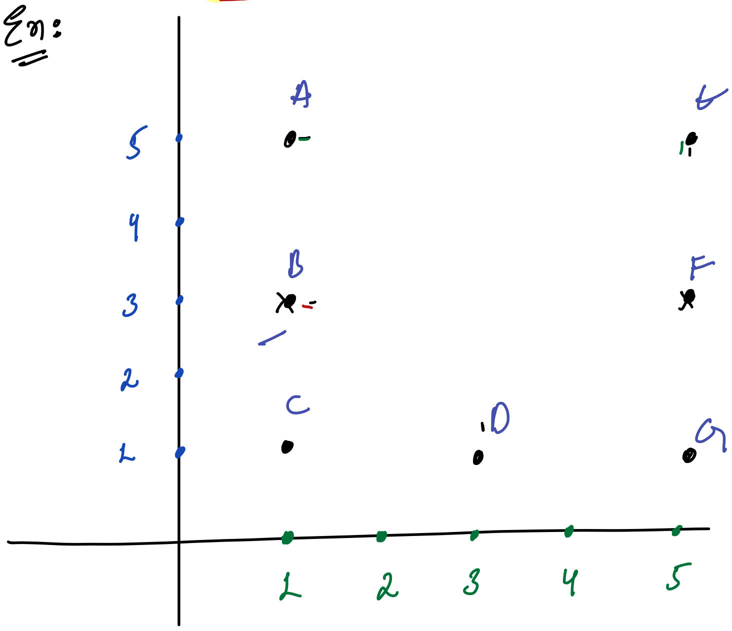
```
c = 0;
i = 0; i < N; i = i+1){        {Center of Right angle triangle}
    (x[i], y[i])  → How many points x = x[i] : Cx = hmx[x[i]]
                  → How many points y = y[i] : Cy = hmy[y[i]]
    c = c + (Cx - 1) * (Cy - 1)
}
return c;
```

PseudoCode :

Given N distinct points in 2D plane, find no: of rectangles such that their sides are parallel to x-axis & y-axis

Note: 4 points should be at 4-corners

Ex:



(1, 1) (5, 3) (1, 3) (1, 5) ✓

BL TR TL BR

(3, 1) (5, 3) 3 3 5, 1 ✗

(1, 3) (5, 5) (1, 5) (5, 3) ✓

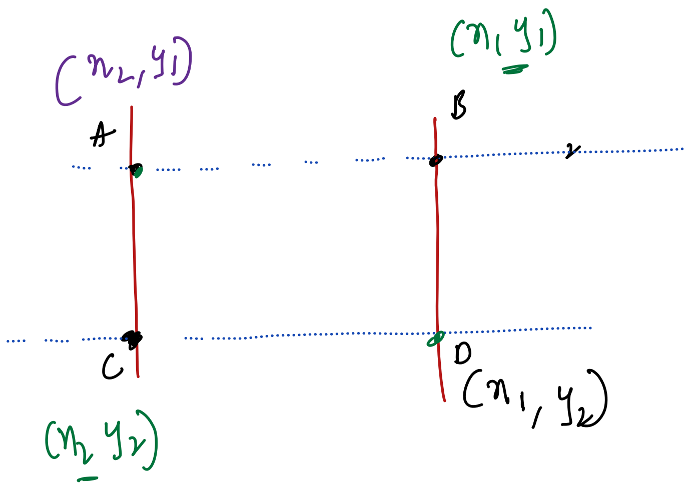
(3, 1) (5, 5) (3, 5) (5, 1) ✗

Idea: For every 4 points, check if they form Rectangle or Not?

$$^{N}C_4 = \frac{N!}{(N-4)! * 4!} = \frac{(N)(N-1)(N-2)(N-3)}{24} \approx O(N^4)$$

Overall: TC: $O(N^4) * \{ O(1) \text{ to check} \}$ TODO

Hint: **Quez:** Min points to find Rectangle, {x-axis, y-axis}



A	D	B	C
(x_1, y_1)	(x_1, y_2)	(x_2, y_1)	(x_1, y_2)
(x_2, y_1)	(x_1, y_2)	(x_1, y_1)	(x_2, y_2)

Idea: If we find 2 diagonal, & if other corners are also present in that can rectange is present

Note: If we find 2 points which are parallel to x-axis or y-axis, {we cannot find Rectangle}

	0	1	2	3
x[ ] :	1	1	5	5
y[ ] :	1	5	1	5

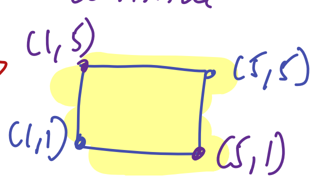
Note: Same rectangle counted twice

Pairs:

<0,1> continue

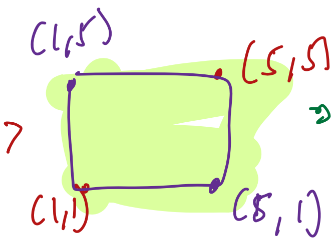
<0,2> continue

<0,3>



$c = c + 1$

<1,2>



$\rightarrow c = c + 1$

<1,3>

<2,3> $\rightarrow$ $c = 2$;

Same rectangle twice

```
int Rectangle(int x[], int y[], int N){
    HashSet<String> hs;
    i=0; i<N; i++){ -> {storing a point in terms of string}
        hs.insert( to_string(x[i]) + "@" + to_string(y[i]));
    }
    c=0;
    i=0; i<N; i++){ -> i^th is finding 1 point
        j=i+1; j<N; j++){ j^th is finding 1 point
            (x1, y1) = (x[i], y[i])   (x2, y2) = (x[j], y[j])
            if( y1==y2 || x1==x2 ){ continue;
                                    we cannot form rectangle }
            // other 2 corners we need to search ?
                    (x2, y1) && (x1, y2)
            String p1 = to_string(x2) + "@" + to_string(y1)
            String p2 = to_string(x1) + "@" + to_string(y2)
            if( p1 is in hs && p2 is in hs ){
                c = c+1
            }
        }
    }
    return c/2
}
```

TC: $O(N^2)$ SC: $O(N)$

Q3) longest substring with all distinct characters?

Given a String S, find the length of the longest substring without repeating characters?

	0	1	2	3	4	5	6	7
S =	a	b	c	a	b	c	d	d

: abc : 3
abcd : 4

	0	1	2	3	4	5	6
S =	s	i	p	p	i	e	r

: pier : 4

	0	1	2	3	4
S =	a	a	a	a	a

: a → valid Substring → 1

Idea:

a) for every substring check if it contains all distinct charaters?

TC: $O(N^3)$
SC: $O(N)$

```
i = 0; i < N; i++) {
    j = i; j < N; j++) {
        Substring → [i j]
        Check if Substring
        S[i j] contain all
        distinct
        TODO: loop & insert all Element in hashset
        if (hs.size() == j-i+1) { ans = max (ans, j-i+1) }
    }
}
return ans;
```

s = a b a c e a f f : ans = 4

(indices: 0 1 … 6)

2 4 3 4 3 2 1 1

: Idea: Take every index as start of a substring & get max substring length which is distinct

// Pseudocode:

```
ans = 0
i = 0; i < N; i++){
    Hashset<char> hs;
    c = 0
    j = i; j < N; j++){
        if( s[j] is in hs) {
            break
        }
        else {
            c++;
            hs.insert(s[j])
        }
    }
    ans = max(ans, c)
}
return ans;
```

TC: O(N^2)

SC: O(N)

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

S = a b c g h i g k l m h a b k o

↑ i (at 8) ↑ j (at 14)

HS :

{ ~~a~~, ~~b~~, ~~c~~, ~~g~~, ~~h~~, ~~i~~, ~~g~~, ~~k~~, l, m, h, a, j, k }

i	j	i	j
0	0		
0	1	1	1
0	2	1	2
0	3	1	3
0	4	1	4
0	5	1	5

0 6 → 1 6 → 2 6 → 3 6 → 4 6 ✓

↓

4 7

↓

4 8

↓

4 9

↓

[4 10]

↓

[5 10] → [5 11] → [5 12] → [5 13]

↓

[6 13] → [7 13] → [8 13] → [9 14]

↓

lour ihden break & return ans

```
i = 0, j = 0, ans = 0

HashSet<char> hs

while( j < N) {
    if( s[j] is not in hs) {
        hs.insert(s[j]) j++;
        ans = max(ans, hs.size())
    }
    else{
        hs.remove(s[i])
        i++;
    }
}
return ans;
```

TC : O(N)

SC : O(N)