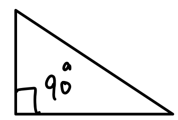
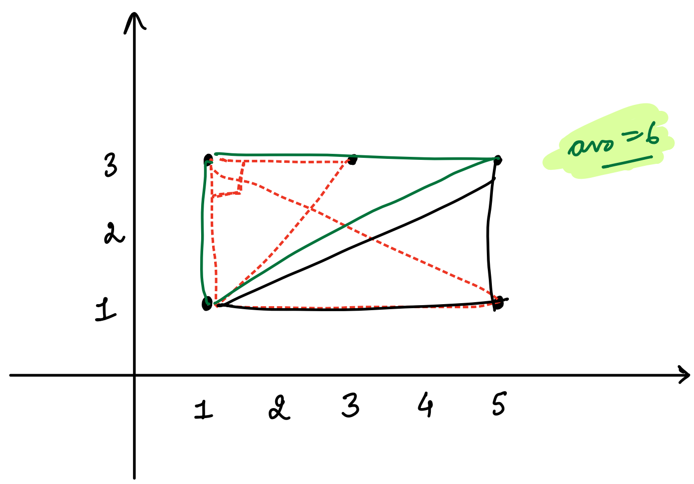
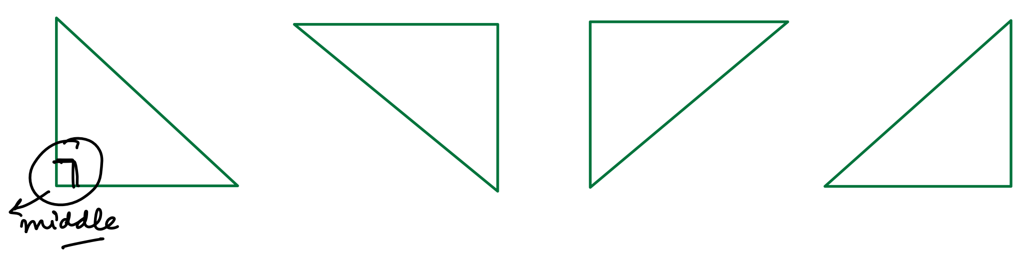
Hashing-2

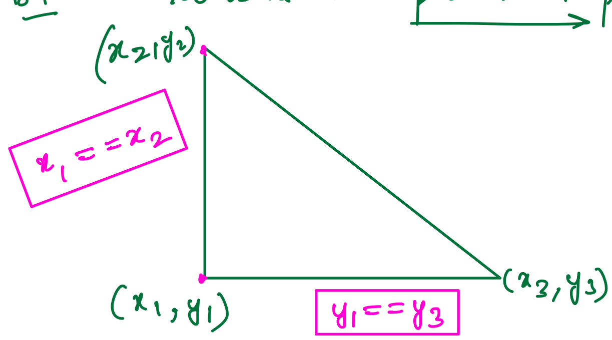
Q Given coordinates of N points on a 2D-plane.
Count no. of right-angle triangles using the given set of points such that two small sides of ◣ should be parallel to x-axis & y-axis.







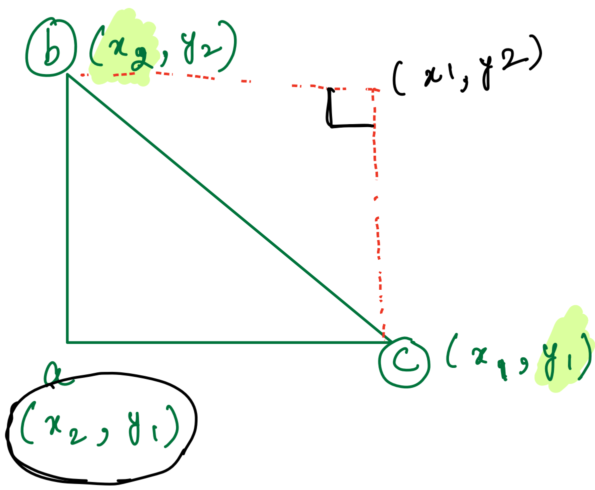
B-F: consider all possible triplets & check if a triplet matches.



```
for( int i=0; i<n; i++)      → middle
  for( j=0; j<n; j++)        → vertical
      if( i==j) continue;
    for( k=0; k<n; k++)      → h2
    {
        if( i==k || j==k) continue;
        if ( x[i] == x[j] && y[i] == y[k])
              cnt++;
    }
```

T.C: $O(n^3)$ → $O(n^2)$

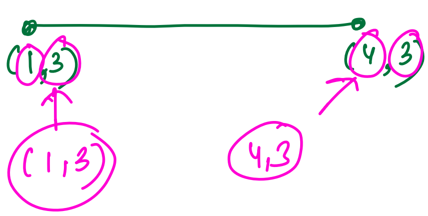
If I have 2 points of the triangle, can you find out the 3rd point?



0,1   1,0

```
for( i=0; i<n; i++)
{
   for( j=i+1; j<n; j++)
   {
      if( x[i] == x[j] || y[i] == y[j])
            continue;
      check( x[i], y[j]) cnt++;
      check( x[j], y[i]) cnt++;
   }
}
```

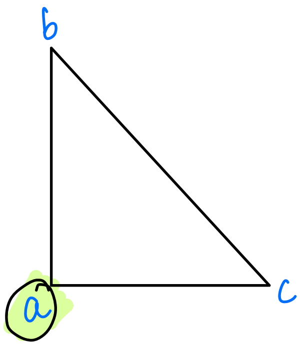
hashset (pair) x_y



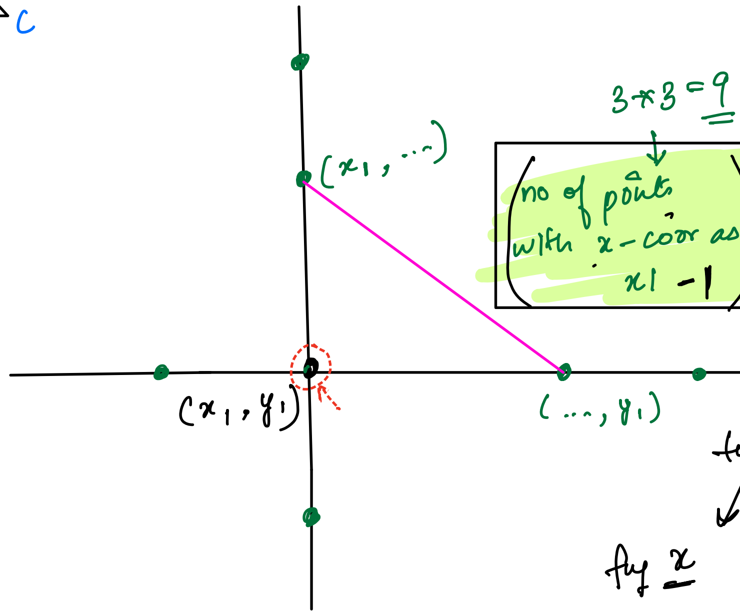
T.C: $O(n^2)$

S.C: $O(n)$

idea:- consider every single point as a middle pt & then find how many △ can be formed with that pt as middle pt



O(n)



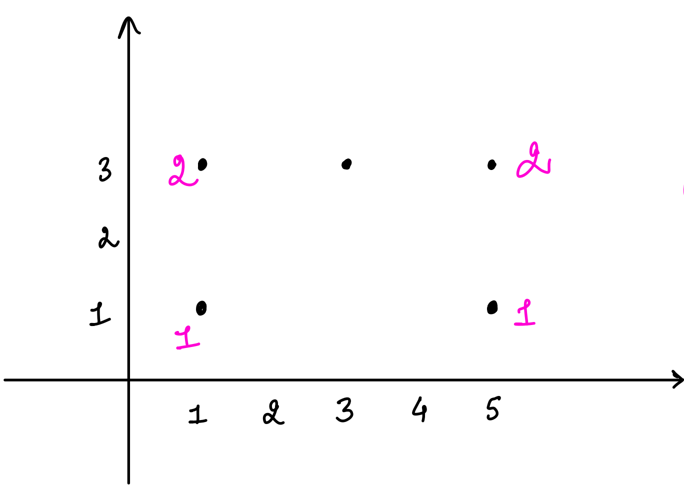
3*3 = 9

(no of points with x-coor as x1 - 1) * (no of point with y cor as y1 - 1)

two hashmaps

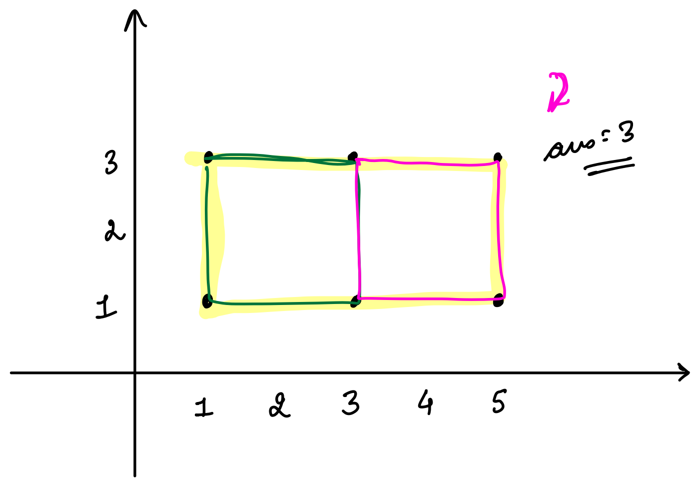
freq x

freqy



6

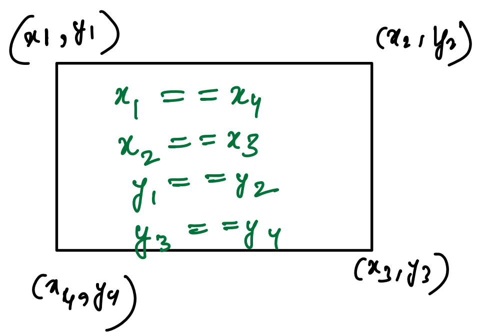
Q Given N points in a 2D-plane.
Find count of rectangles we can form s.t. sides are parallel to x-axis and y-axis

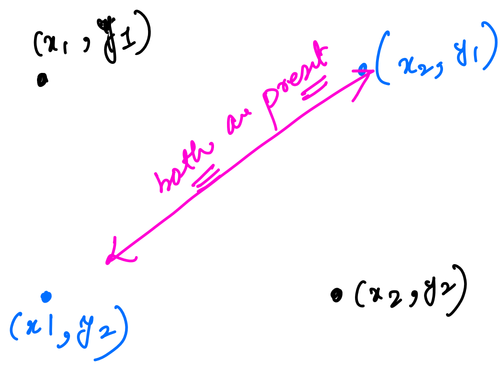


2

ans = 3

B.F :- consider all quadruples





$\Rightarrow$ $O(n^2)$

Q given a string s & an integer K.

Find if s can be rearranged to form a string which is concatenation of K equal strings.

s = "abbbbabba"
K = 3

⇓

abb abb abb
bba bba bba

__ __ __ K buckets

hashmap → $\boxed{\text{count of every char \% k == 0}}$

$n \leftarrow S =$ a b c b a b x a

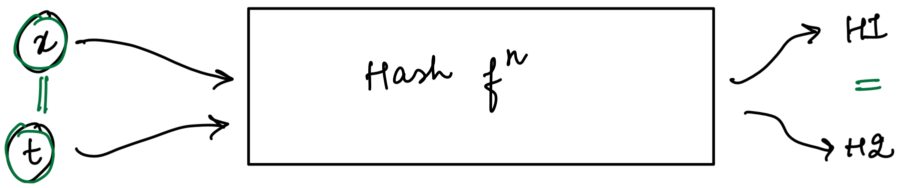
$k \leftarrow T =$ c b a

Find if T is present in S as a substr

pattern matching

substring → subarray

BoF :- $n*k$



H1 = H2

a → 97

a b c ⇒ 97+98+99

sum of ascii values

~~0 1 2~~ . . .

1 2 3

a   aa   aaa   aaaa . . .

a b c d e f ⇒ 21

b c a d e f ⇒ 21

6!

⇑ collisions

$\sum s[i]$

$\sum s[i] * i$

1 2 3 4 5 $\Rightarrow$ $5*10^0 + 4*10^1 + 3*10^2 + 2*10^3 + 1*10^4$

base, no of dig.

$\sum \cancel{s[i] * i^a}$

charac = 26

$\boxed{\sum s[i] * p^i}$ $\rightarrow$ $p = 26$

mod

aa

ba

$\boxed{\sum s[i] * p^i \;\%\; m}$ $\rightarrow$ $0 - m-1$

very

roll

10^9+7

$m = 10^9+7$

0 ---- h2 ---- h1 ---- m-1

$s_1 \rightarrow h(s_1)$ $\rightarrow 0$

$s_2 \rightarrow 1/m \approx 10^{-9}$

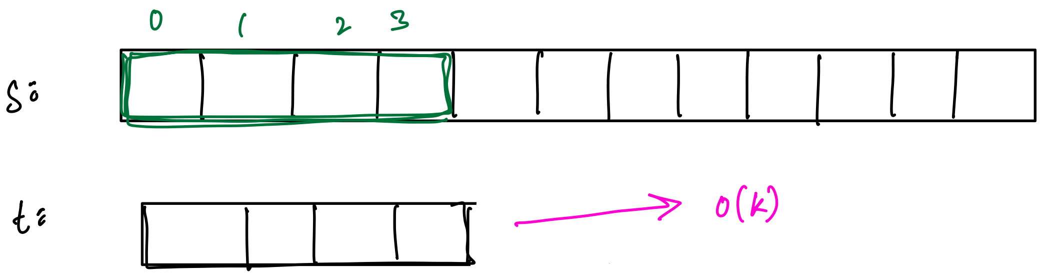
s_3 $\quad 2/m \approx 10^{-9}$

$\vdots$

s_{n+1} $\quad n/m$ $\quad 10^5 / 10^9 \approx 10^{-4}$ $\quad n \approx 10^5$

0.0001

plagrism

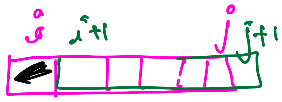


→ $O(k)$

$$h(t) = \underline{h(s[0-3])}$$ match

$\downarrow$

$O(k)$



$$\sum s[i] * p^i \ \% \ m$$

$$h(s[i-j]) = (s[i] + s[i+1]*p + s[i+2]*p^2 + s[i+3]*p^3 \ldots$$
$$s[j]*p^{j-i}) \% m$$

$\Downarrow$

$$\boxed{\sum_{k=i}^{j} s[k] * p^{k-i} \ \% \ m} = H \quad - (1)$$

$$h(s[i+1-j+1]) = (s[i+1] + s[i+2]*p + s[i+2]*p^2 \ldots$$
$$s[j]*p^{j-i-1} + s[j+1]*p^{j-i}) \% \text{mod}$$

$$H' = \left(\frac{H - s[i]}{P} + s[j+1]*p^{j-i}\right) \% m \quad - (2) \quad \rightarrow O(1)$$

$c = 26$
$p = 31$

$p^{-1} \% m$ pow fermat

$$\boxed{p^{m-2} \% m \rightarrow \log(m)}$$ precompute

Patten $O(n+k)$ Rabin Karp Algorithm (?) Z algorithm

$$\boxed{\frac{H - \text{outgoingvalue}}{p} + \text{incoming} * p^{k-1} \ \% \ m}$$

pattern in a text → Rabin karp