Today's Content:

✓ → pair sum K

✓ → Distinct elements in Subarray K.

Q Given an array N. check if their exists a pair (i,j)

S.T $arr[i] + arr[j] = K$ & $i != j$ { i,j are indexes }

	0	1	2	3	4	5	6	7	8	9
arr[] =	8	9	1	-2	4	5	11	-6	7	5

K = 11

i	j
4	8

$\underset{4}{arr[i]} + \underset{7}{arr[j]} = 11$

True

K = 6

i	j
2	5
0	3
2	9

1 + 5 = 6

8 - 2 = 6

1 + 5 = 6

True

K = 10

i	j
5	9
1	2

5 + 5 = 10

9 + 1 = 10

True

K = 22

i	j
6	6

11 + 11 = 22

False

Idea1   check all pair

$TC: O(N^2)$
$SC: O(1)$

```
for( i=0; i < N; i++)
{
    for( j=0 ; j<N; j++)
    {
        if( i != j && arr[i] + arr[j] == K)
            return True
    }
}
return False
```

```
01   12   23  34
02   13   24  35
03   14   25
04   15
05            45
```

```
for( i=0; i < N; i++)
{
    for( j=i+1 ; j<N; j++)
    {
        if( arr[i] + arr[j] == K)
            return True
    }
}
return False
```

$TC: O(N^2)$
$SC: O(1)$

First element = arr[0]

Second element = K - arr[0]

i^{th} → First element

K - arr[i]

K = 11

$$\left\{ \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & \\ 8, & 9, & 1, & -2, & 4, & 7 \end{array} \right\}$$

1. Create hashset with all elements
2. For every i, search K - arr[i] in the hashset.

$$\left\{ \begin{array}{c} 8 \\ 9 \\ 1 \\ -2 \\ 4 \\ 7 \end{array} \right\}$$

First element	Second element	Yes/NO
8	3	NO
9	2	NO
1	10	NO
-2	13	NO

ans = True

4	7	Yes

Ex = {5, 1}

K = 10

{5
1}

False

First elem	K-a Second	Yes / No
5	5	Yes.

Ex {5, 1, 5}

True

1. Create hashmap, by storing frequency.

2. for every i, search k - arr[i]

arr[i] == K-arr[i]   freq > 1

arr[i] != K-arr[i]   f > 0

{2, 4, 2, 6,
7, 3}

K = 14

{
< 2, ~~1~~ 2 >
< 4, 1 >
< 6, 1 >
< 7, 1 >
< 3, 1 >
}

First	Second	Yes/NO
2	12	NO
4	10	NO
2	12	NO
6	8	NO
7	7	NO
3	11	NO

return False

```
bool    checkpair ( arr[], int K)
{
        int n = arr.length
        Hashmap < int, int >  hm
        for ( i=0; i<N; i++)                              N
        {
            If ( hm.containsKey( arr[i]))
                        hm.put ( arr[i] , hm.get(arr[i]) + 1)
            else
                        hm.put( arr[i], 1)
        }

        for( i=0; i<n; i++)
        {
            int   a = arr[i]                             N
                  b = K - a
            If (a != b)
            {
        }
}
```

```
{
    if( hm.containsKey(b ))
                        return True
}
else
{
    if( hm.get(b) >1)
                        return True
}
return False
```

TC : O(N)

SC : O(N)

TC : O(N)
SC : O(N)

>

TC : O (N log N)
SC : O (1)

Break

10:07 10:15

Q. Given an array N. Cal no of distinct elements in every subarray of size K.

$$arr[6] = \begin{array}{cccccc} 0 & 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 3 & 8 & 3 & 4 \end{array}$$

$K = 4$

[0,3]	4
[1,4]	3
[2,5]	3

$s = 0, \quad e = K-1$

$1, \quad K$

$2, \quad K+1$

$3, \quad K+2$

$\vdots$

$N-K, \quad N-1$

# of subarrays = $N-K+1$

```
s=0, e= K-1
while( e < N)
{
    Hashset <int> hs
    for( i=s ; i≤e ; i++)     hs.put (arr[i])
    print ( hs.size())
    s++
    e++
}
```

TC : $(N-K+1)*K$

$K=1 \quad (N-1+1)\,1 = N$

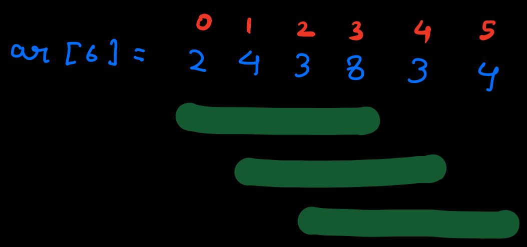
$K=N \quad (N-N+1)\,N = N$

$K = \frac{N}{2}$ $\quad (N - \frac{N}{2} + 1)(\frac{N}{2})$

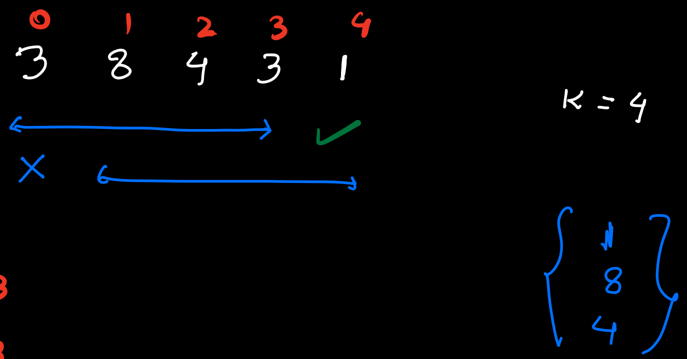
$$(\frac{N}{2} + 1)(\frac{N}{2}) = \frac{N^2}{4} + \frac{N}{2} \quad O(N^2)$$

TC: $O(N^2)$

SC: $O(K)$



2 Sliding window on hashset



0-3	3
1-4	3

Issue: deleting a number in hashset, deletes all the occurences.

Idea 3 Hashmap + Sliding window.

	0	1	2	3	4	5	6	7	8	9	
arr[ ] =	2	4	3	8	3	9	4	9	4	10	K = 4

0 - 3	4
1 - 4	3
2 - 5	3
3 ~ 6	4
4 - 7	⋮
5 - 8	
6 - 9	

{
<4, 1>
<9, 1>
<3, 1>
<8, 1>
}

Pseudo Code

```
Hashmap <int,int> hm
for( i=0 ; i≤ K-1 ; i++)
{
    if ( arr[i] in hm)        update freq
    else        hm.put( arr[i], 1)
}
print ( hm.size())
s = 1 , e = K
while ( e < N)
{
    // Remove    arr [s-1]
    // add       arr [e]
```

```
Decrease freq of arr[s-1]
if ( hm.get ( arr[s-1]) == 0)
{   remove element

if ( arr[e] in hm )     freq++
else     mp.put ( arr[e], 1)

print( hm.size() )
s++
e++
```

TC: O(N)

SC: O(K)

Doubts

$$pre[e] - pre[s-1] = 0$$

$$pre[e] = pre[s-1]$$