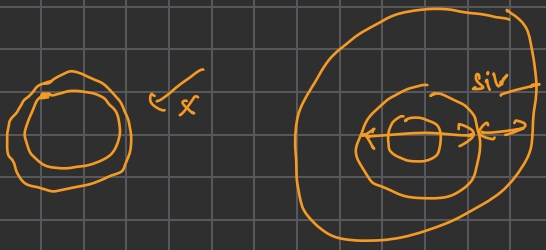

x
siv

2pass O(n)
O(n)
O(n)

totalwater = 0 2 6 9 13 18 19

single pass

Rightmax = 0 3 4

Trapping

Leftmax = 0 2 4

Leftside check

{

if(Leftmax > arr[i])

totalwater = Leftmax - arr[i];

}

else

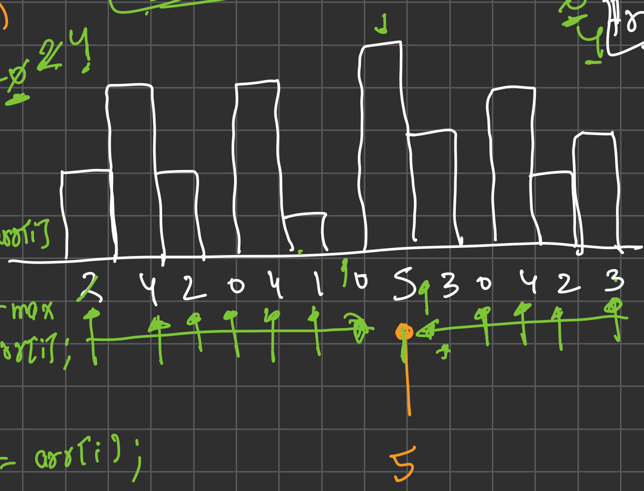
{

Left Max = arr[i];

}

left++;

}



min(Leftmax, Right max) - arr[i];

min(Leftmax, Rightmax) - arr[i];

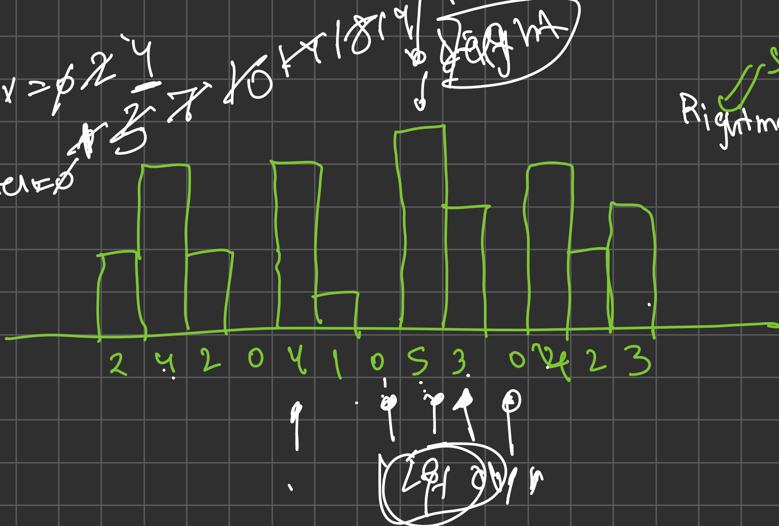
maximum
Right

Solve

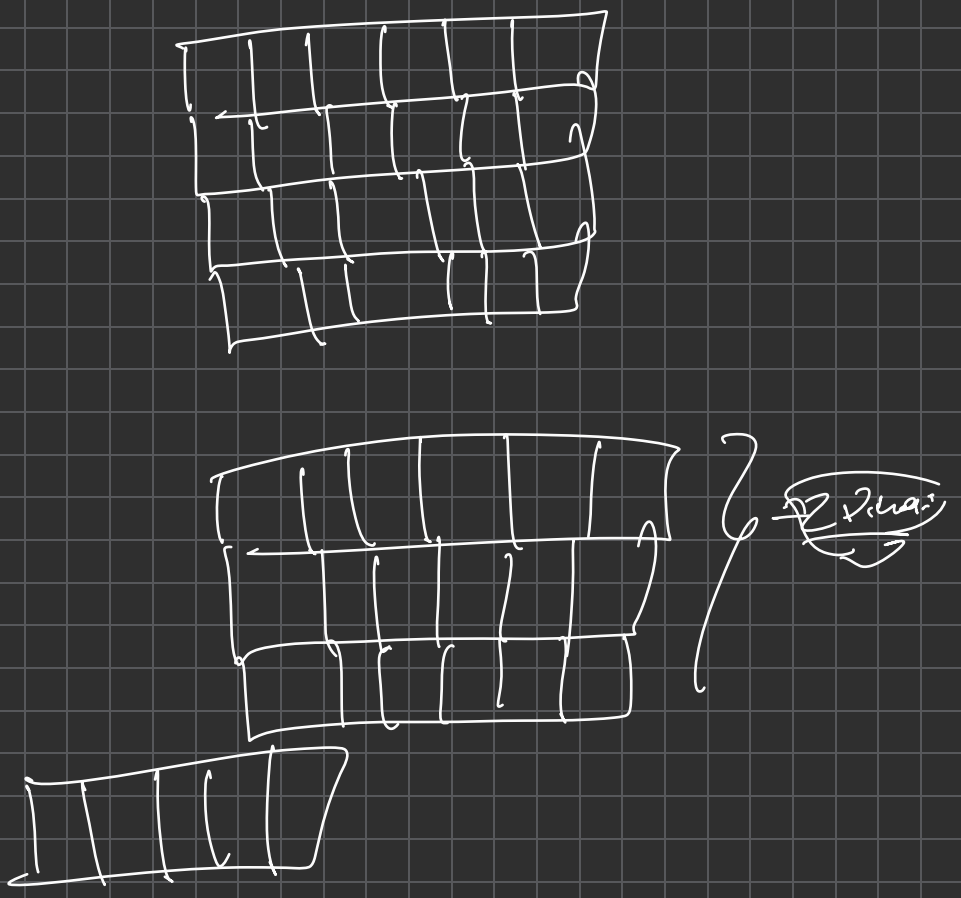
Leftmax = 0 2 4

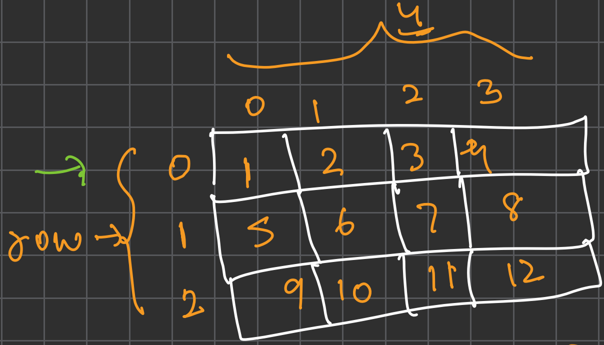
totalwater = 0 2 5 7 10 14 18 19

Rightmax = 0 3 4



2d array — 2 dimensional array





3 rows
4 column

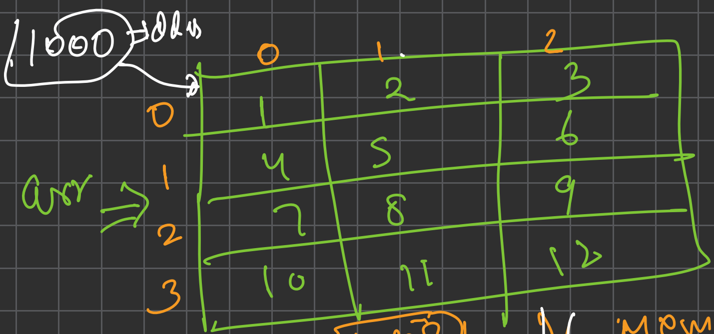
arr [1][2] ✓
(7)

int arr[3][4] = { {1, 2, 3, 4}, {5, 6, 7, 8},
{9, 10, 11, 12} };

print t value kaise kara sakte hu;

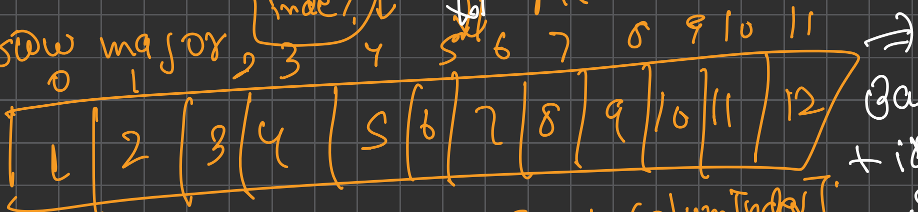
arr[0][0]
✓ arr[0][1]
arr[0][2]

```
for (i=0 ; i<3 ; i++)
{
  for (j=0 ; j<4 ; j++)
  { cout << arr[i][j]; }
}
```



4 → row
3 → colum
4x3 = 12

Memory Location



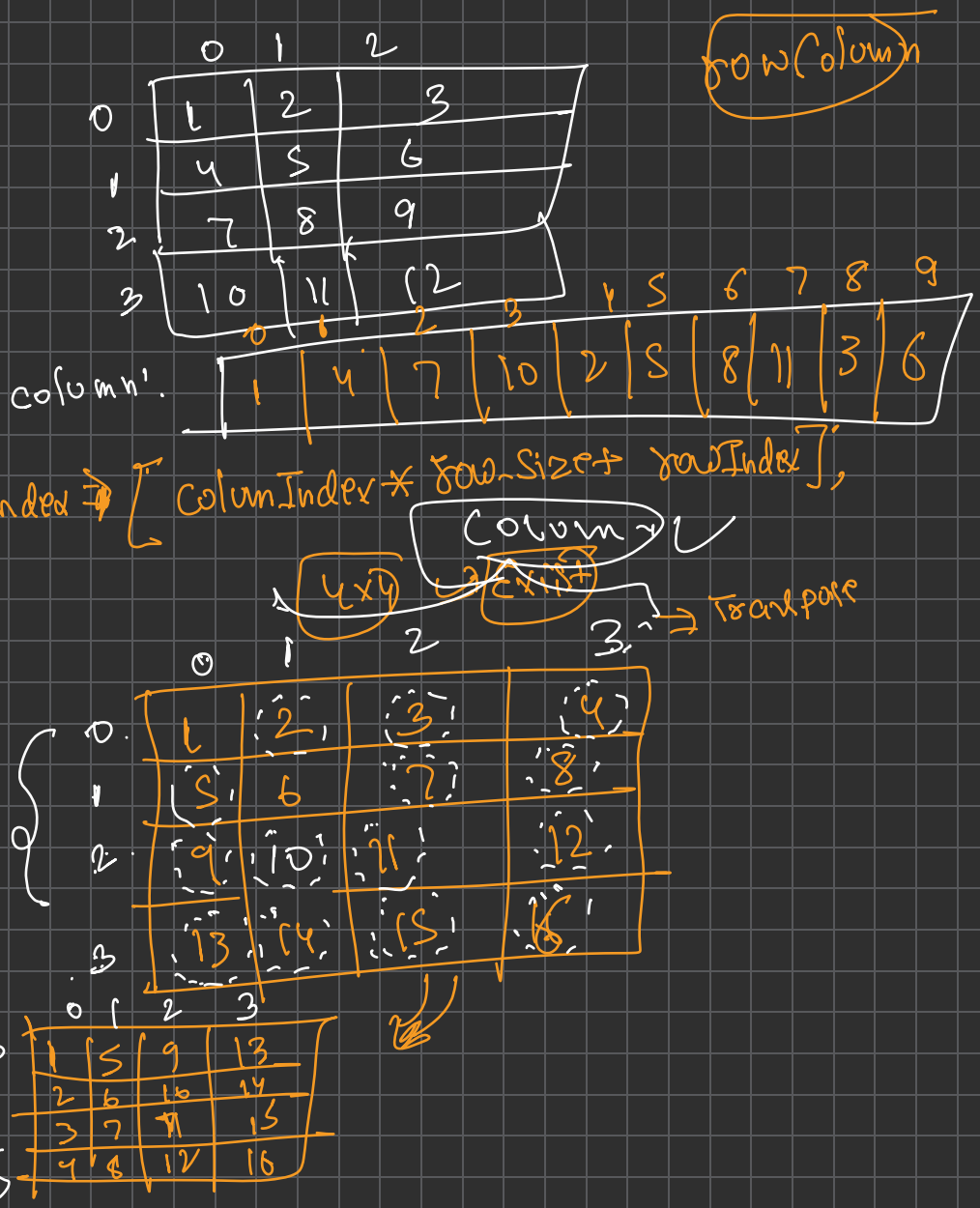
index ⇒ { rowIndex × ColumSize + ColumIndex },

$2 \times 3 + 1 = 7^{th}$

address = Base address + ( ) * sizeofData

= 1000 + 40

= **1040**


row Column
index = [ColumIndex * row_Size + rowIndex];
Column
4x4
2x4
Transpose
row