

# Binary Search

Observation  
searching

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

$d = 6$

```
while ( i <= j ) {  
    mid = ( i + j ) / 2 ;  
    if ( a[mid] < d )  
        i = mid + 1  
    else if ( d < a[mid] )  
        j = mid - 1  
    else  
        index = mid  
        break ;  
}
```

<sup>0</sup> 1    <sup>1</sup> 5    <sup>2</sup> 10    <sup>3</sup> 15    <sup>4</sup> 22    <sup>5</sup> 33    <sup>6</sup> 40    <sup>7</sup> 42    <sup>8</sup> 55    <sup>9</sup> 66



$j < i$

$$mid = (i + j) / 2$$

if (arr[mid] < d)  
 floor = arr[mid]  
 i = mid + 1

if (d < arr[mid])  
 ceil = arr[mid]  
 j = mid - 1

else

not exist

$d = 34$

ceil = ~~42~~ 40

floor = ~~22~~ 33

1 5 10 15 22 33 40 42 55 66 34

fin 2  
ceil  
floor

1

5

10

15

22

33

i j  
40

mid  
42

53

66

$j < i$

$$mid = (i + j) / 2$$

if (ar[mid] < d)  
floor = ar[mid]  
i = mid + 1

if (d < ar[mid])  
ceil = ar[mid]  
j = mid - 1

else  
ceil = floor = ar[mid]  
break

not exist

d = 40

ceil = ~~42~~ 40

floor = ~~22~~ 33 40

0 1 2 3 4 5 6 7 8

1 1 2 2 2 4 4 4 5 5 6 7 7 8

[illegible]

$d = 3$

d

2  
exist  $\rightarrow$  last 2 2  
last 4

 $[2, 4]$ 

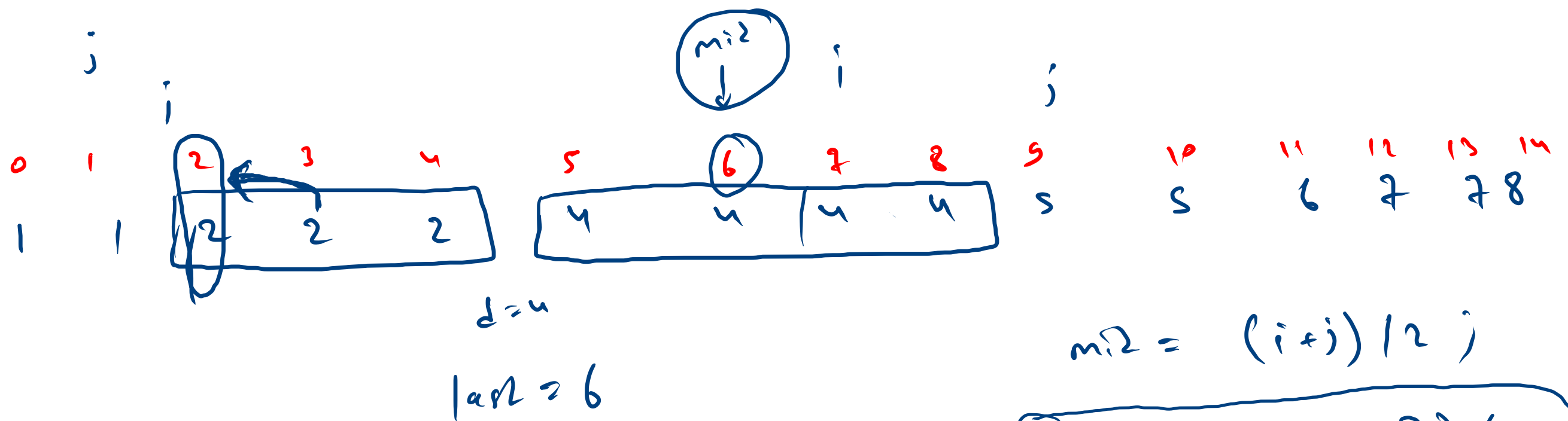
## 2. binary search

Done

Last

3 → 2M -1  
not exn 2as2 -1

$$[-1, -1]$$



$$mid = (i + j) / 2$$

if ( $d < ar[mid]$ )  
 $j = mid + 1$

else if ( $ar[mid] < d$ )  
 $i = mid + 1$

else  
 $low = mid$   
 $j = mid - 1$

first  $\neq$  2

→ not exist

marks = 10

6

int marks[] = new int[3];

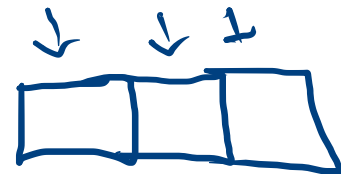
int marks[][] = new int[2][3];

s11 =

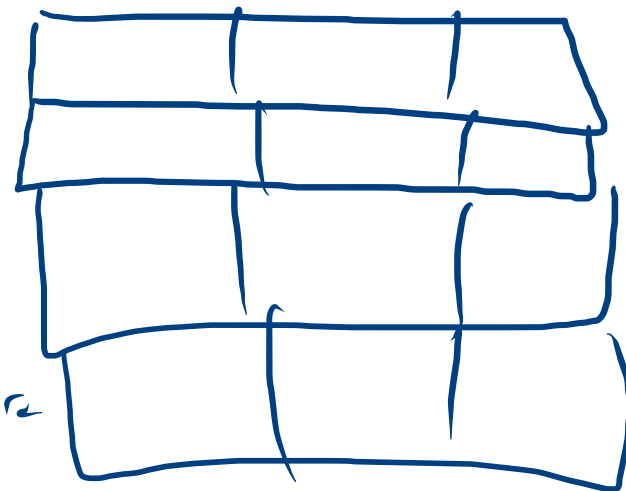
s12 =

s23 =

s24 =



p com hindi



marks[i]

i[marks]

	0	1	2	3
0		10		
1			5	
2				

marks[i][j]

row col

[i]

row

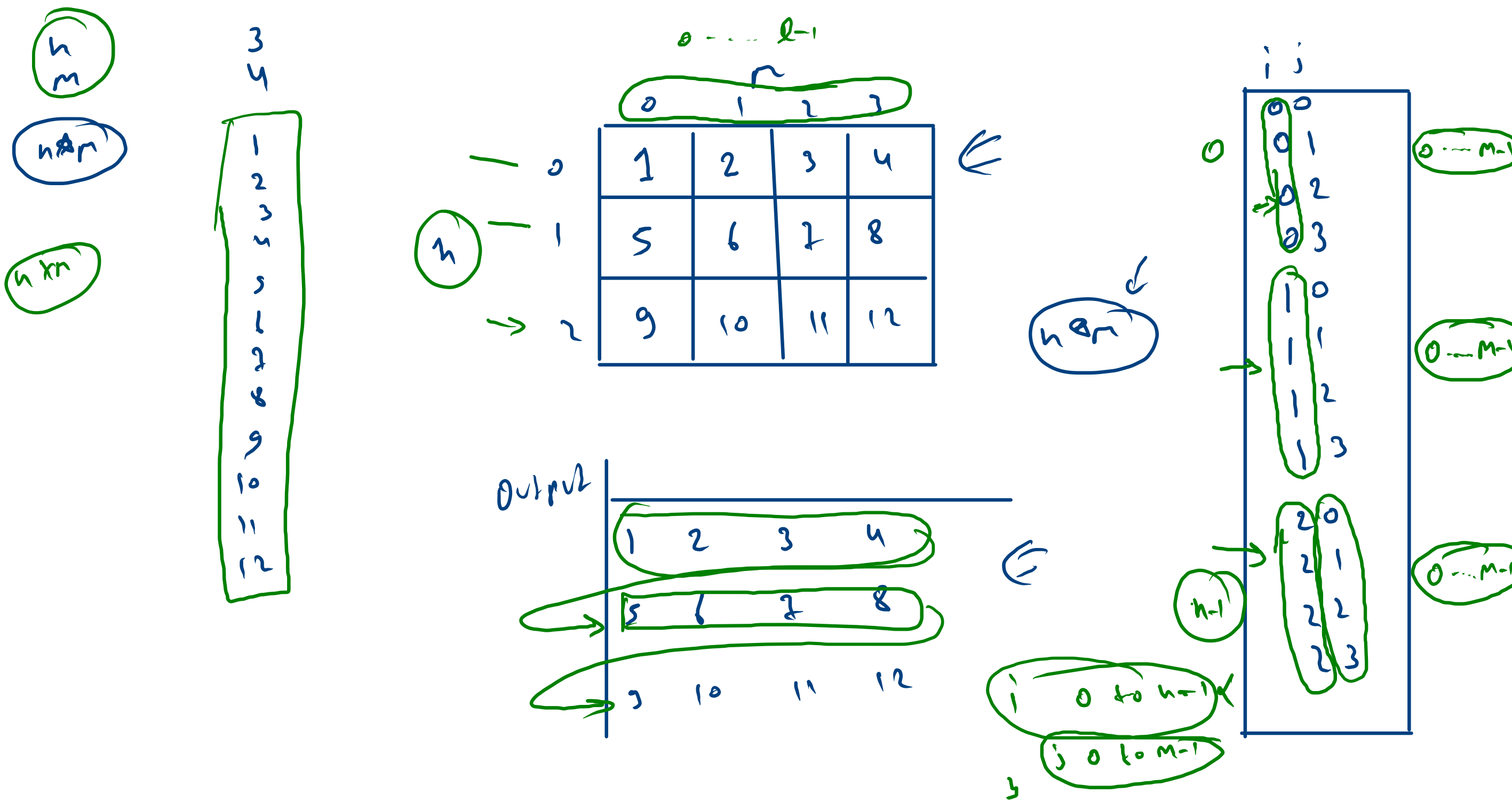
[j]

col

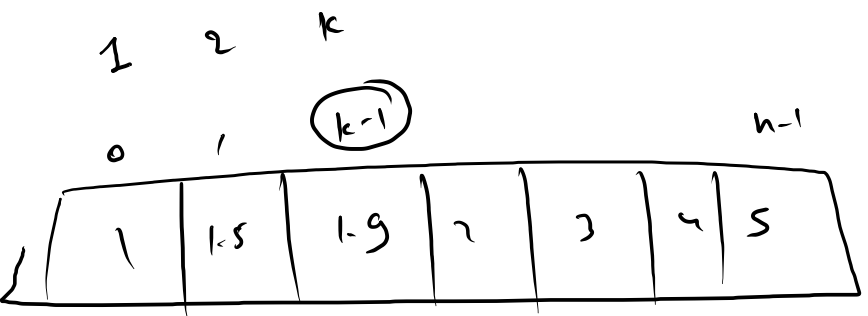
marks[1][2] = 5

marks[0][1] = 2 \* marks[1][2]

= 10



long l() = new long[4]



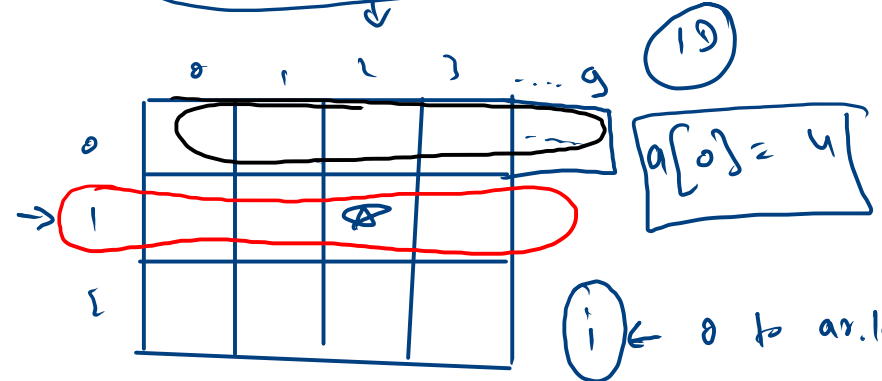
Arrays.sort(ar)

h = k

pa

h \* m \* int + h \* pa

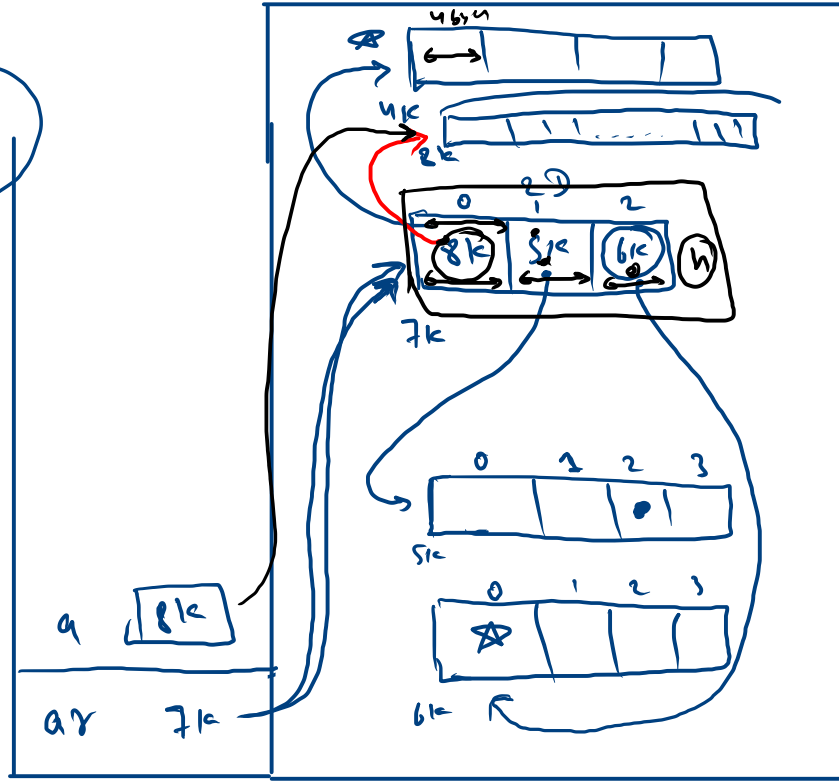
mark 2  
 int ar[][] = new int[5][4];  
 int a[] = new int[10];  
 ar[0] = a;



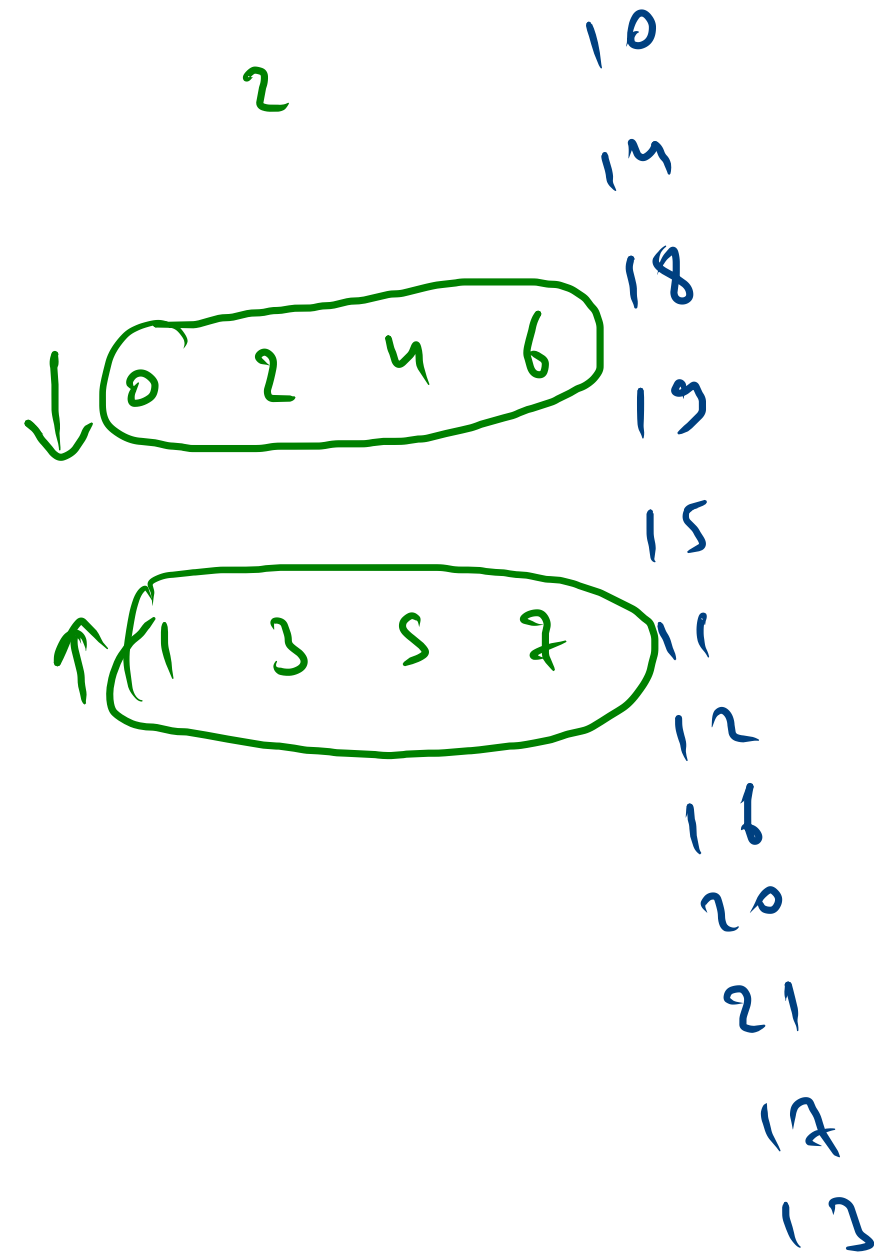
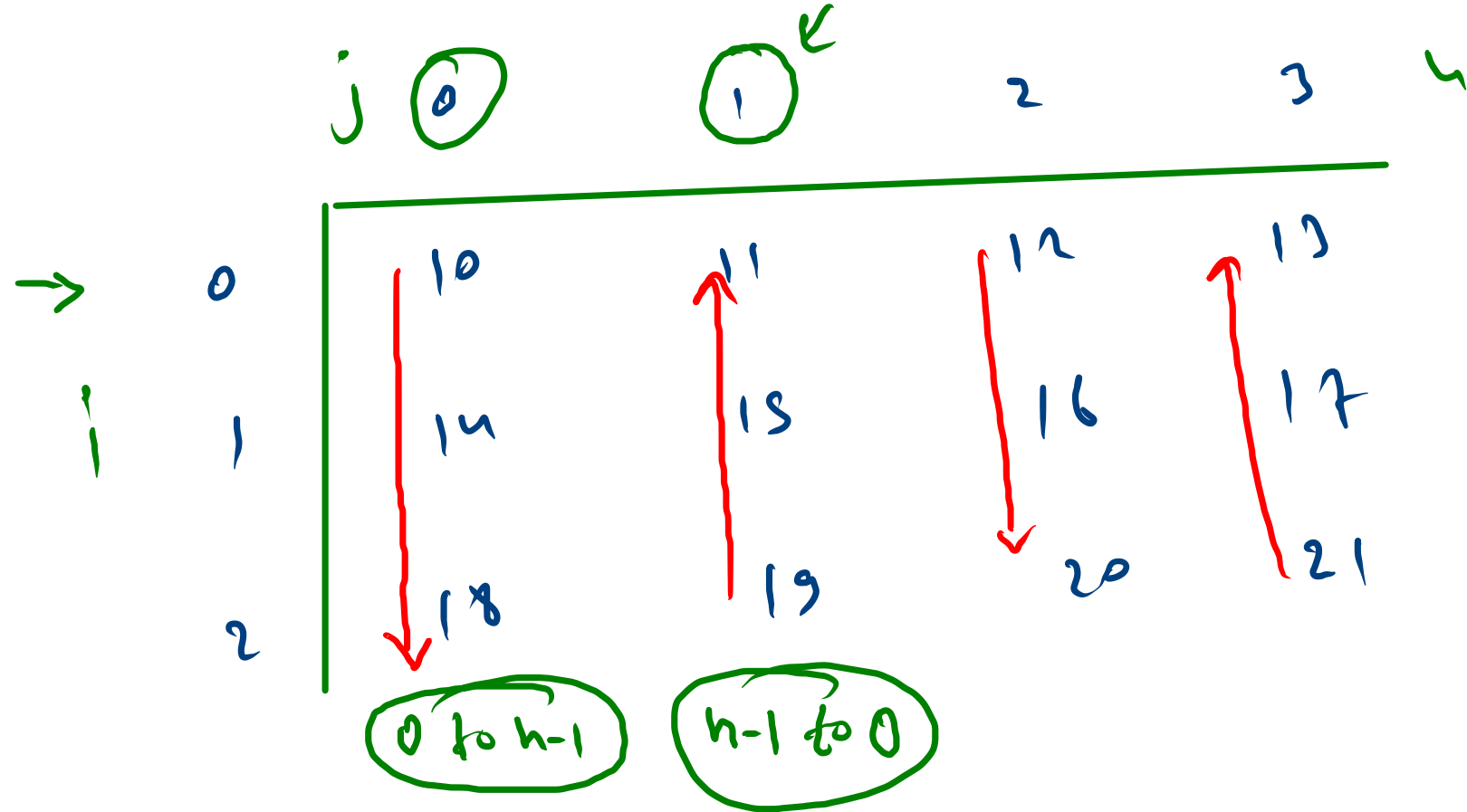
i ← 0 to ar.length - 1  
 j ← 0 to ar[i].length - 1

1D  
 2D

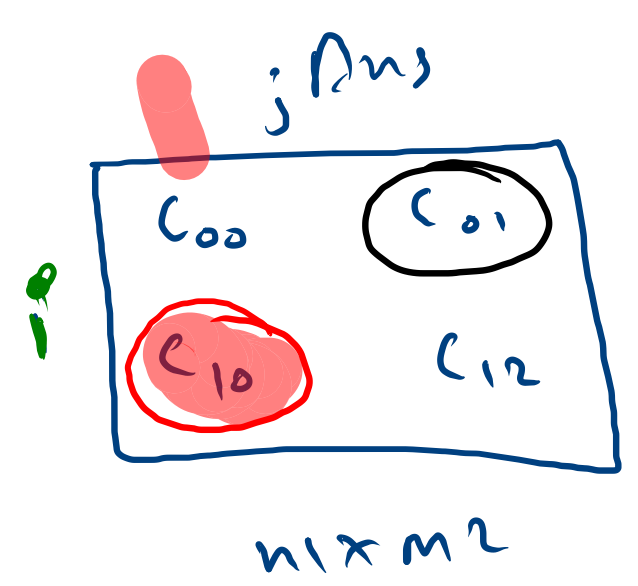
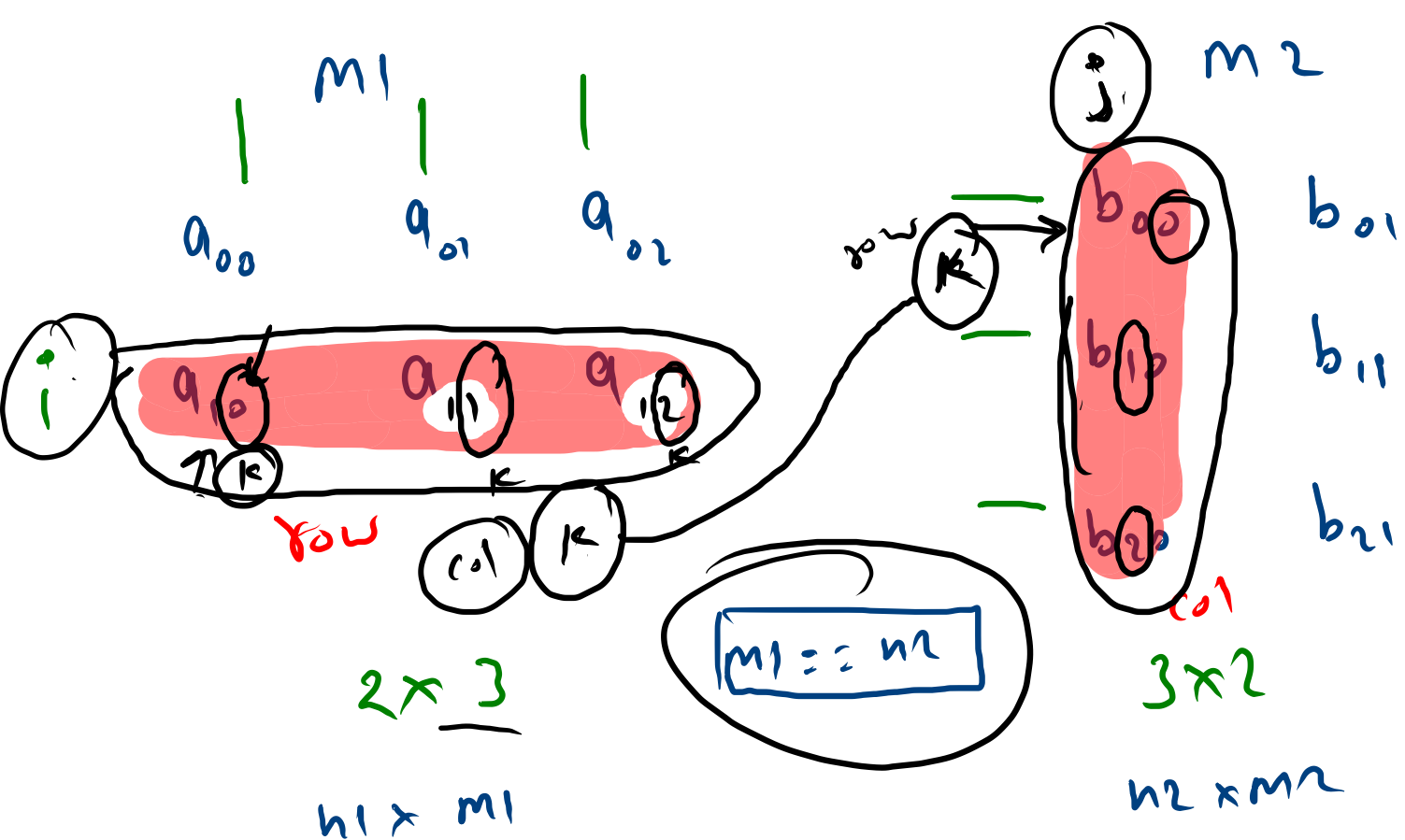
long → 4 bytes







$j = 0$  to  $n-1$   
 if ( $j \% 2 == 0$ )  
      $i \leftarrow 0$  to  $n-1$   
 else  $n-1$  to 0



$$C_{10} = a_{10} \times b_{00} + a_{11} \times b_{10} + a_{12} \times b_{20}$$

Diagram illustrating the calculation of  $C_{10}$  (row 1, column 0) as the sum of products of corresponding elements from row 1 of matrix A and column 0 of matrix B:

- $a_{10} \times b_{00}$
- $a_{11} \times b_{10}$
- $a_{12} \times b_{20}$

$i$  0 to  $n1 - 1$   
 $j$  0 to  $m2 - 1$   
 $k$  0 to  $m1 - 1$   
 $A[i][k] \times B[k][j]$   
 $ans[i][j] = \text{sum};$

