2D Array

$$int [][] arr = new int [3][3];$$

Heap

Stack

4K   14k 0   1   2

0 | 14k | →
1 | 24k | → 24k
2 | 34k | → 34k

arr = 4k
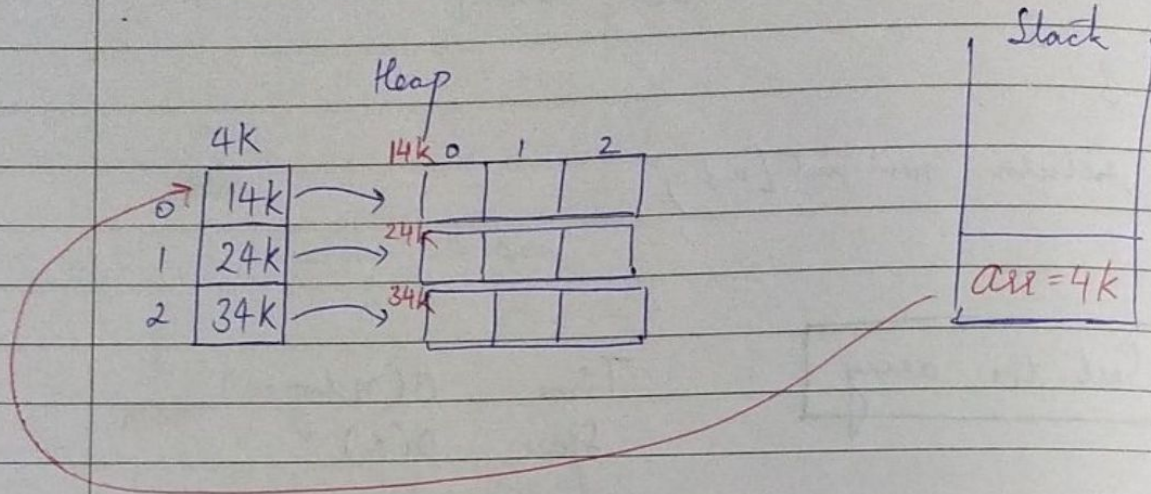
Print 2D array

```
for( int i = 0; i < arr. length ; i++) {
    for ( int j=0; j < arr [i]. length ; j++) {
        syso ( arr [i][j] + " " );
    }
}
```

* int [][] arr = new int [ ][3] ; // ERROR
* int [][] arr = new int [3][ ] ;
  arr [0] = new int [4] ;
  arr [2] = new int [3] ;

syso (arr [0]) ;   // prints the ref. address
syso (arr [1]) ;   // null
syso (arr [2]) ;   // prints the ref. address

syso (arr [3]) ;   // index out of bounds exception

syso (arr[1][2]) ;   // Null pointer exception

OR
int[][] arr = { {11, 12, 13}, {21, 22, 23}, {31, 32, 33} }

Take Input of 2D array & print it

```
int [][] arr = takeInput ();
display (arr);
private static int [][] takeInput () {
    Scanner sc = new Scanner (System. in );
    S.O.P. ( " Enter no. of rows");
    int rows = s. nextInt ();
    int[][] arr = new int [rows][];

    for ( int row =0 ; row < rows ; row ++) {
        S.O.P. ( " Enter no. of cols for row" + row);
        int cols = s. nextInt ();
        arr [row] = new int [cols];
        for ( int col = 0 ; col < cols ; col ++) {
            S.O.P. ("Enter the value for row" + row +
                    " and col" + col );

            arr [row][col] = s. nextInt ();
        }
    }
    return arr;
}

private static void display ( int [][] arr ) {
    for ( int row = 0 ; row < arr. length ; row ++) {
        for ( int col = 0 ; col < arr[row]. length ; col ++){

            S.O. print ( arr[row][col] + " " );
        }
        S.O.P() ;
    }
}
```

Wave Print

```
11    12    13    14
21    22    23    24
31    32    33    34
41    42    43    44
```
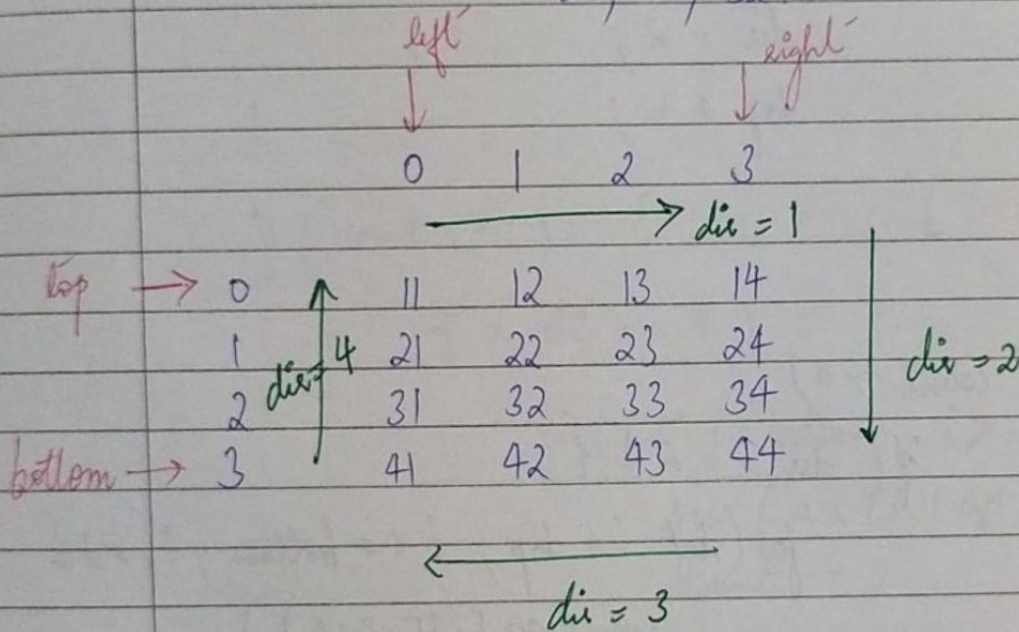
```java
public static void printRowWise (int[][] arr) {
    for (int i =0; i < arr.length; i++) {
        if (i % 2 == 0) {
            for (int j =0; j < arr[i].length; j++) {
                S.O.P. (arr[i][j] + "  ");
            }
        }
        else {
            for (int j = arr[i].length -1; j >=0; j--) {
                S.O.P. (arr[i][j] + "  ");
            }
        }
    }
    S.O.P. (" END ");
```

Spiral Print

	0	1	2	3
0	11	12	13	14
1	21	22	23	24
2	31	32	33	34
3	41	42	43	44

O/p : 11, 12, 13, 14, 24, 34, 44, 43, 42, 41, 31, 21, 22, 23, 33, 32.

left   right

	0	1	2	3

$dir = 1$

top →	0	11	12	13	14
	1	21	22	23	24
	2	31	32	33	34
bottom →	3	41	42	43	44

$dir = 4$ $dir = 2$

$dir = 3$

6 parameters

$top = 0;$

$bottom = arr.length - 1;$

$left = 0;$

$right = arr[top].length - 1;$

$16 \Leftarrow count = (bottom + 1) * (right + 1);$

$dir = 1;$

```
while ( left <= right && top <= bottom ) {
    if ( count > 0 ) {
        if ( dir == 1 ) {
            for ( int i = left ; i <= right ; i++ ) {
                syso ( arr [top][i] );
                count --;
            }
            dir = 2;
            top ++;
        }
    }
    if ( count > 0 ) {
        if ( dir == 2 ) {
            for ( int i = top ; i <= bottom ; i++ ) {
                syso ( arr [i][right] );
                count --;
            }
            dir = 3;
            right --;
        }
    }
}
```

```
if ( count > 0 ) {
    if ( dir == 3 ) {
        for (int i = right ; i >= left ; i--) {
            syso ( ars [bottom][i] );
            count -- ;
        }
        dir = 4 ;
        bottom -- ;
    }
}

if ( count > 0 ) {
    if ( dir == 4 ) {
        for (int i = bottom ; i >= top ; i--) {
            syso ( ars [i][left] ) ;
            count -- ;
        }
        dir = 1 ;
        left ++ ;
    }
}

syso ( "END" );
```