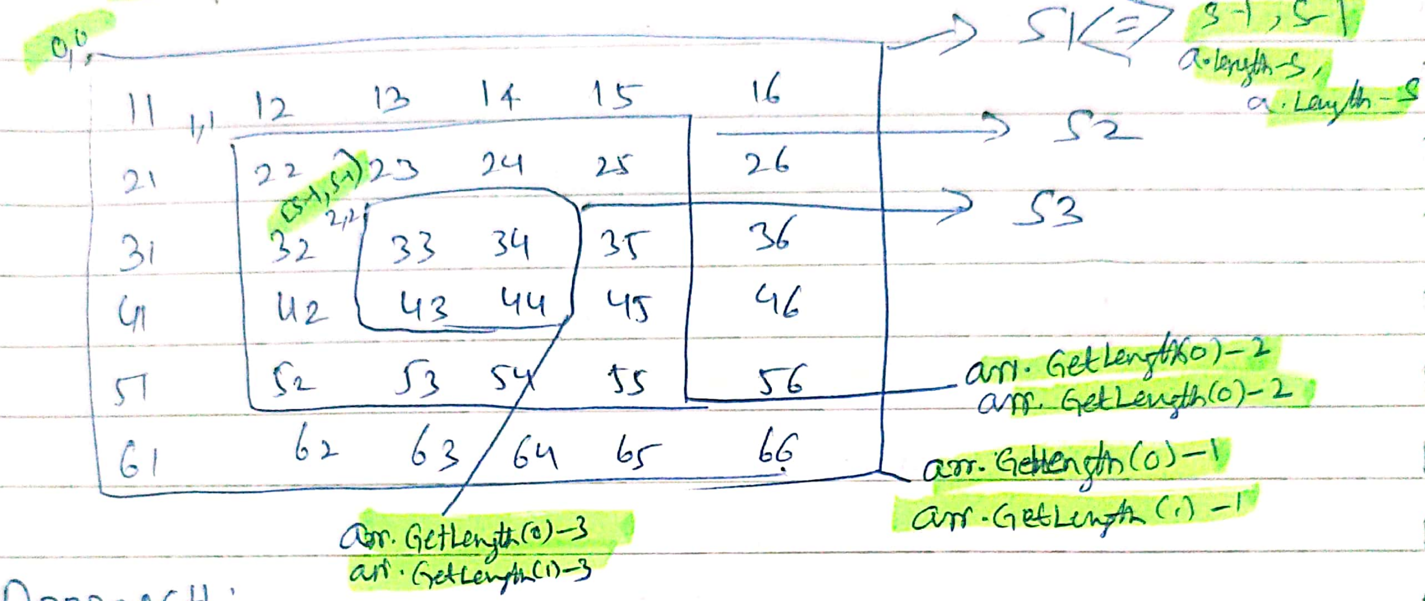
Shell rotate : 2D matrix : Anti clockwise rotate Shell for an "R" No of Rotations



11	12	13	14	15	16
21	22	23	24	25	26
31	32	33	34	35	36
41	42	43	44	45	46
51	52	53	54	55	56
61	62	63	64	65	66

APPROACH :

(1) MOVE SHELL INTO A 1D ARRAY

(2) ROTATE 1D ARRAY

(3) MOVE ROTATED 1D ARRAY BACK TO SHELL

```
int s = Console.WriteLine();
int r = Console.WriteLine();

rotateShell(arr, s, r);
```

```
Public static void rotateshell(int[,] arr, int s, int r){
    int[] oned = filloned fromshell(arr, s);
      rotate(oned, r);
    fill shellfromOned(arr, s, oned);
}

Public static void rotate(int[] oned, int r){
    r = r % oned.Length;
    if(r<0){
        r = r + oned.Length; // important
    }
    ~~reverse(oned, 0 oned.Length~~
    ~~reverse(Oned, , oned.Length-1)~~
    reverse(Oned, 0, Oned.Length-r-1);//Part 1
    reverse(Oned, Oned.Length-r, Oned.Length-1);//part2
    reverse(Oned, 0, Oned.Length-1); [Full Reverse]
}

Public static void reverse(int[] oned, int li, int ri){
    While(li<ri){
        int temp = oned[li];
        Oned[Li] = oned[ri];
        Oned[ri] = temp;
        li++;
        ri--;
    }
}
```

```
Public static int[] fillonedfromshell(int[,] arr, int s){

        int minr = s-1;
        int minc = s-1;
        int maxr = arr.GetLength(0) -s;
        int maxc = arr.GetLength(1) - s;

  int sz = 2*(maxr-minr + maxc-minc);
     int[] oned = new int[sz];
        int idx=0;
     // lw
     for(int i=minr, j=minc; i<=maxr; i++)
     {
         oned[idx] = arr[i,j];
         idx++;
     }


     // bw
     for(int i=maxr, j=minc+1; j<=maxc; j++){
                oned[idx]= arr[i,j];
                idx++;
            }


       // rw
     for(int i=maxr-1, j = maxc; i >= minr; i--){
                oned[idx]=arr[i,j]
                 idx++;
       }
```

max c - minc+1

lw+bw+rw+tp

maxr-minr+1

maxr-minr+1

2*(maxr-minr+1)-4

2*(maxc-minc+1) - 4

```
  // tw
  for(int i=maxr, j= maxc-1; j>=minc+1 ; j--){
          oned[idx] =arr[i, j]
            idx++;
     }
   return oned;
 }


Public static void fillshellFromOned(int[,] arr, int s, int[] oned)
 {

      int minr = s-1;
      int minc = s-1;
      int maxr= arr.GetLength(0)-s;
      int maxC = arr.GetLength(1)-s;

      int idx=0;
      // lw
      for (int i =minr, j=minc; i<=maxr; i++){
            arr [i, j]=Oned[idx];
              idx++;
       }

       // bw
        for (int i =maxr, j=minc+1; j <=maxc; j++){
   arr [i, j] = Oned[idx];
           idx++;
        }
```

```
//rw
for (int i=maxr-1, j=maxc ; i>=minr ;i--){
            arr[i,j]=Oned[idx];
            idx++;
    }

  // tw
for(int i=minr, j=maxc-1 ; j>=minc+1 ; j--){
            arr[i,j]=oned[idx];
            idx++;
    }
}
```