Week 2

1. Find key in O(n) time.

```
#include <stdio.h>

int main()
{
    int n;
    scanf("%d",&n);
    int arr[n][n];
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            scanf("%d",&arr[i][j]);
        }
    }
    int target;
    printf("Enter target\n");
    scanf("%d",&target);
    int i=0;
    int j=n-1;
    int flag=0;
    while(i<n && j>=0)
    {
        if(arr[i][j]==target)
        {
            printf("Yes\n");
            flag=1;
            break;
        }
```

```
        if(arr[i][j]>target)
        {
            j--;
        }
        else{
            i++;
        }
    }
    if(flag==0)
    {
      printf("No\n");
    }
}
```

2. Max 1's in a row

```
#include <stdio.h>

int main()
{
    int n;
    scanf("%d",&n);
    int arr[n][n];
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<n;j++)
        {
            scanf("%d",&arr[i][j]);
        }
    }
    int target;
    printf("Enter target\n");
```

```
    scanf("%d",&target);
    int i=0;
    int j=n-1;
    int max=-1;
    while(i<n && j>=0)
    {
       if(arr[i][j]==1)
       {
        Max=i;
          J--;


       }
       else{
           i++;
       }
    }
    Printf(“%d”,max);
}
```

3. Rotate array in clock wise direction

```
#include <bits/stdc++.h>
using namespace std;

void rotateElementsInMatrixClockwise(int n,int m, int mat[4][4])
{
```

```
int row = 0, col = 0;
int prev, curr;
int lastRow= n;
int lastCol= m;
while (row < lastRow && col < lastCol)

{
   if (row + 1 == lastRow || col + 1 == lastCol)  break;

   prev = mat[row + 1][col];
   //for the first row which is in bounds
   for (int i = col; i < lastCol; i++)
   {
      curr = mat[row][i];
      mat[row][i] = prev;
      prev = curr;
   }
   row++;
   //for the last column which is in bounds
   for (int i = row; i < lastRow; i++)
   {
      curr = mat[i][lastCol-1];
      mat[i][lastCol-1] = prev;
      prev = curr;
   }
   lastCol--;
   //for the last row which is in bounds
   if (row < lastRow)
   {
      for (int i = lastCol-1; i >= col; i--)
      {
         curr = mat[lastRow-1][i];
```

```
            mat[lastRow-1][i] = prev;
            prev = curr;
        }
    }
    lastRow--;
    //for the first row which is in bounds
    if (col < lastCol)
    {
        for (int i = lastRow-1; i >= row; i--)
        {
            curr = mat[i][col];
            mat[i][col] = prev;
            prev = curr;
        }
    }
    col++;
  }

  for (int i=0; i<4; i++)
  {
    for (int j=0; j<4; j++)
      cout << mat[i][j] << " ";
    cout << "\n";
  }
}


int main()
{
  int a[4][4] = { {1,  2,  3,  4},
                  {5,  6,  7,  8},
                  {9,  10, 11, 12},
```

```
                                {13, 14, 15, 16} };

        rotateElementsInMatrixClockwise(4,4,a);
    return 0;
}
```