Issue Date: 11-Apr-2022

Objective:

- Learn to calculate the Time/Space equation.
- To get your hands dirty: It's good for you ☺.

Task – 1: ***Time step counting of some dummy code snippets.***

A

```
sum=0;
for (i=1; i<n; i++)
{
     for (j=1; j<i*i; j++)
     {
          if (j%i ==0)
          {
               for (k=0; k<j; k++)
                    sum++;
          }
     }
}
```

B

```
for ( int h=0; h<n ; h+=2 )
{
     for( int p=1; p<=n*n ; p*=3 )
     {}
     for ( int q = 2; q*q<=n ; q++ )
     {}
}
```

Task – 2:

Find Time and Space Equation for the following functions:

Part-A: Remove Duplicates in the received array.

```
void removeDuplicates(int * arr, int & N)
{
     for (int i=0; i<N; i++)
     {
          for (int j=i+1;j<N; j++)
          {
               if (arr[i]==arr[j])
               {
                    for (int k=j;k<N-1; k++)
                         arr[k] = arr[k+1];
                    j--;
                    N--;
               }
          }
     }
}
```

Sample Run for your understanding:

```
int main()
{
     int arr[] = {10, 5, 8, 10, 20, 6, 9, 5, 5, 8, 19};
     int size = 11;
     removeDuplicates(arr, size);
     cout<<"Array Size = "<<size<<'\n';//prints 7
     for (int i=0; i<size; i++)
          cout<<arr[i]<<", ";//prints 10, 5, 8, 20, 6, 9, 19
     return 0;
}
```

Part-B: Returns a pascal triangle of height N.

```
int** createPascalTriangle(int n)
{
     int cnt=0;
```

```
    int** ptrToTri = new int* [n];
    for (int i = 0; i < n; i++)
        ptrToTri[i] = new int[n];
    for (int i = 0; i < n; i++)
    {
        for (int j = 0; j < n; j++)
        {
            if (i >= j)
            {
                if ( (i == j || j == 0) )
                {
                    ptrToTri[i][j] = 1;
                }
                Else
                {
                    int x = ptrToTri[i - 1][j] +
ptrToTri[i - 1][j - 1];
                    ptrToTri[i][j] = x;
                }
            }
        }
    }
    return ptrToTri;
}
```

Sample Run for your understanding

```
int main()
{
    int height=5;
    int * * tri = createPascalTriangle(height);
    for (int i=0; i<height; i++)
    {
        for (int j=0; j<=i; j++)
        {
            cout<<tri[i][j]<<'\t';
        }
        cout<<'\n';
    }
    return 0;
}
/*
//Prints
1
1    1
1    2    1
1    3    3    1
1    4    6    4    1
*/
```

Task – 3:

You are given an array. You have to write a program that will print all the leaders in the array. An element is leader if it is greater than all the elements to its right side. Rightmost is always a leader. Do it in O(N) time and O(1) space.

For Example: in array {6, 7, 4, 3, 5, 2}, leaders are 7, 5 and 2.

Task – 4:

The input is an N order matrix of integers. Each individual row is increasing from left to right and each individual column is increasing from top to bottom.

Give a O(N) time worst case algorithm that decides if a number X is in the matrix or not?