30 Days Leetcode Challenge - Day 11

import java.util.HashSet;

public class leetday11

{

public static void longestConsecutiveSequenceOptimal(int arr [])

{

int n = arr.length;

int count = 0;

int longest = 1;

HashSet<Integer> set = new HashSet<>();

for(int i=0;i<n;i++)

{

set.add(arr[i]);

}

for(int imPresent : set)

{

if(!set.contains(imPresent-1))//illai

{

count = 1;

int x = imPresent;

while(set.contains(x+1))

{

x=x+1;

count = count +1;

}

longest = Math.max(longest,count);

}

}

System.out.println(longest);

}

public static void matrixToZeroOptimal(int arr[] [],int n,int m)

{

int col0 = 1;

for(int i=0;i<n;i++)

{

for(int j=0;j<m;j++)

{

if(arr[i][j]==0)

{

arr[i][0]=0;

if(j!=0)

{

arr[0][j]=0;

}

else

{

col0 = 0;

}

}

}

}

for(int i=1;i<n;i++)

{

for(int j=1;j<m;j++)

{

if(arr[i][j]!=0) // in the previous we had row[i] = 1 or col[j] = 1 which is the same as arr[i][j] !=0

{

if(arr[i][0]==0 || arr[0][j]==0)

{

arr[i][j]=0;

}

}

}

}

if(arr[0][0]==0)

{

for(int j=0;j<m;j++)

{

arr[0][j]=0;

}

}

if(col0==0)

{

for(int i=0;i<n;i++)

{

arr[i][0]=0;

}

}

for(int i=0;i<n;i++)

{

for(int j=0;j<m;j++)

{

System.out.print(arr[i][j]+ " ");

}

System.out.println();

}

}

}