# Experiment-1.2

## Student Name: Nabha Varshney UID: 20BCS4995

**Branch: CSE Section/Group: 20BCS-DM-704 (A)**

## Semester: 6th Date of Performance: 22th Feb 2023 Subject Name: Competitive Coding II Subject Code: 20CSP- 351

**Aim** – To implement the concept of Arrays, Queues and Stack and Linked List.

## Objective-

* The objective is to build problem solving capability and to learn the basic concepts of data structures.
* The implementation of arrays, queues which shows and brushes up the concept of 1D, 2D arrays and can be solved through various approaches.
* The implementation of removing duplicates from the sorted list was introduced.

## Jump Game II

<https://leetcode.com/problems/jump-game-ii/>

**Code –**

class Solution {

public:

    int jump(vector<int>& nums) {

        int len=nums.size()-1;

        int curr=-1,next=0,ans=0;

        for(int i=0;next<len;i++)

        {

            if(i>curr){

                ans++;

                curr=next;

            }

        next=max(next,nums[i]+1);

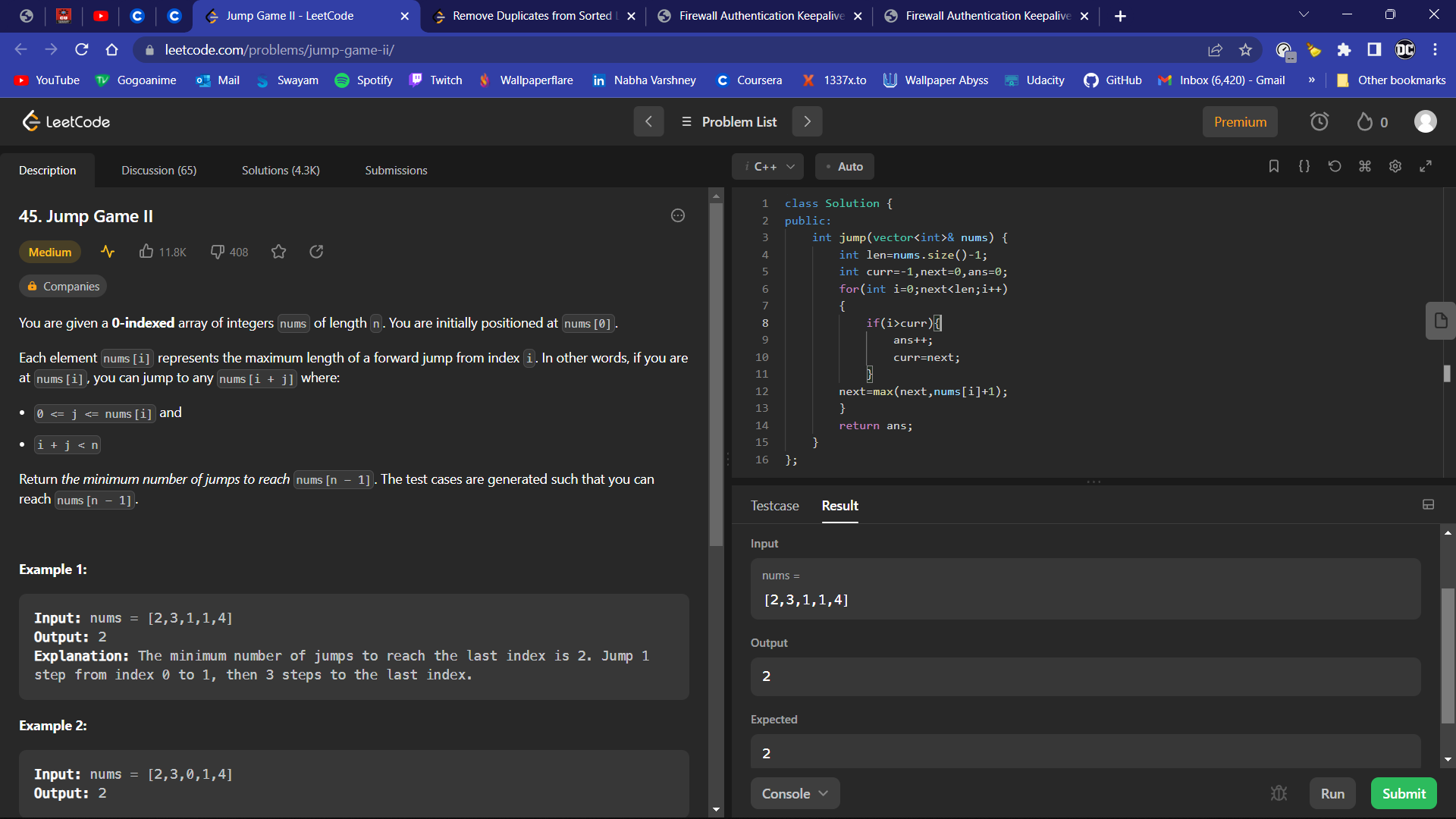
        }

        return ans;

    }

};

## Output -



1. **Remove the duplicate elements from list**

<https://leetcode.com/problems/remove-duplicates-from-sorted-list-ii/>

**Code -**

class Solution {

public:

    ListNode\* deleteDuplicates(ListNode\* head) {

        if(head==0||head->next==0)return head;

        int flag=0;

        ListNode\*prev=0,\*cur=head,\*nex=head->next,\*dummy=0;

           prev=new ListNode;

            dummy=prev;

        prev->next=head;

        while(nex!=0)

        {

            if(cur->val==nex->val)

            {

                prev->next=nex;

                cur=prev->next;

                nex=cur->next;

                flag=1;

            }

            else if(flag==1)

            {

                 prev->next=nex;

                cur=prev->next;

                nex=cur->next;

                flag=0;

            }

            else

            {

                prev=prev->next;

                cur=cur->next;

              nex=nex->next;

            }

        }

         if(flag==1)

            {

                prev->next=0;

            }

        return dummy->next;

    }

};

## Output -

