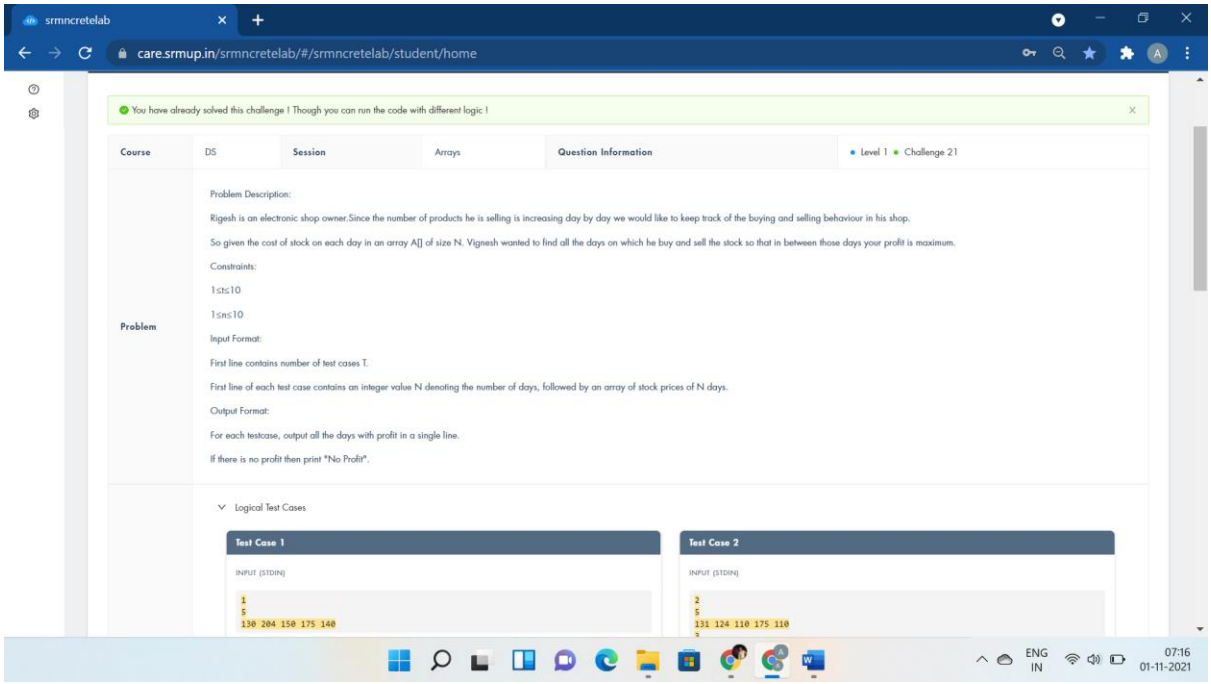
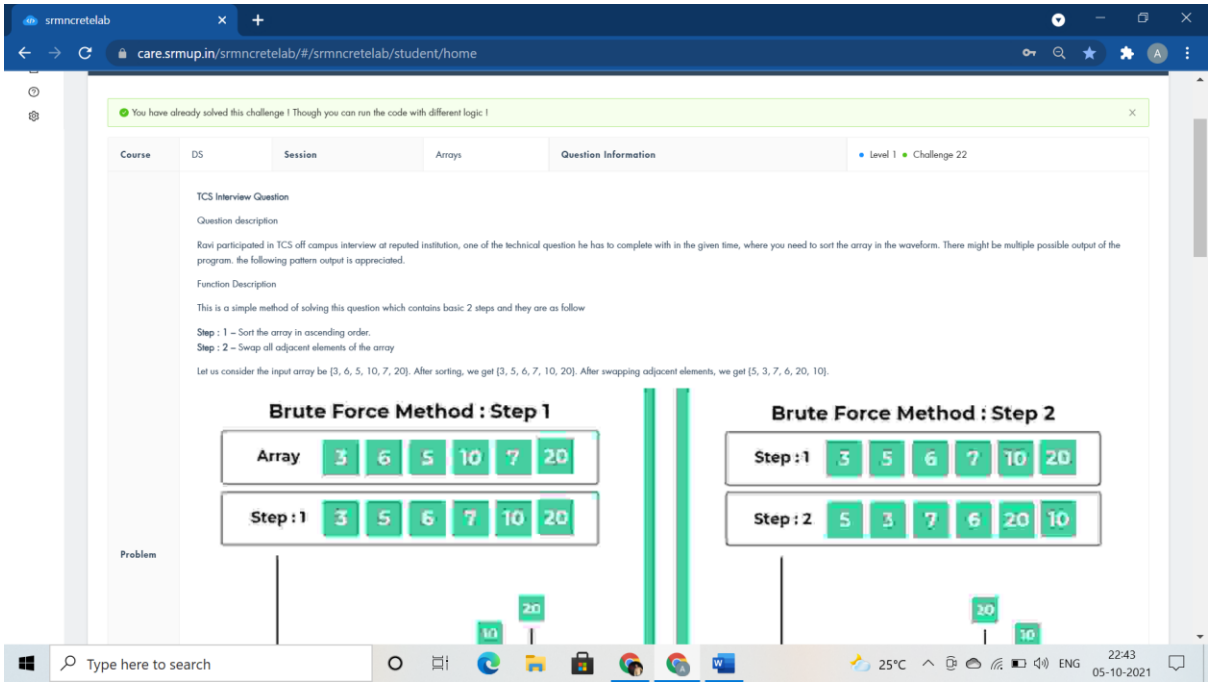
# DSA ARRAYS




```
#include <stdio.h>

struct interval
{
   int buy;
   int sell;
};


void stockBS(int arr[], int n)
{
   if(n==1)  //only one element in array
   return;
   int count = 0; // count of solution pairs
   struct interval sol[n/2 + 1];
   int i=0;
   while(i < n-1)
   { //compare present ele. with next
      while((i < n-1) && (arr[i+1] <= arr[i]))
```

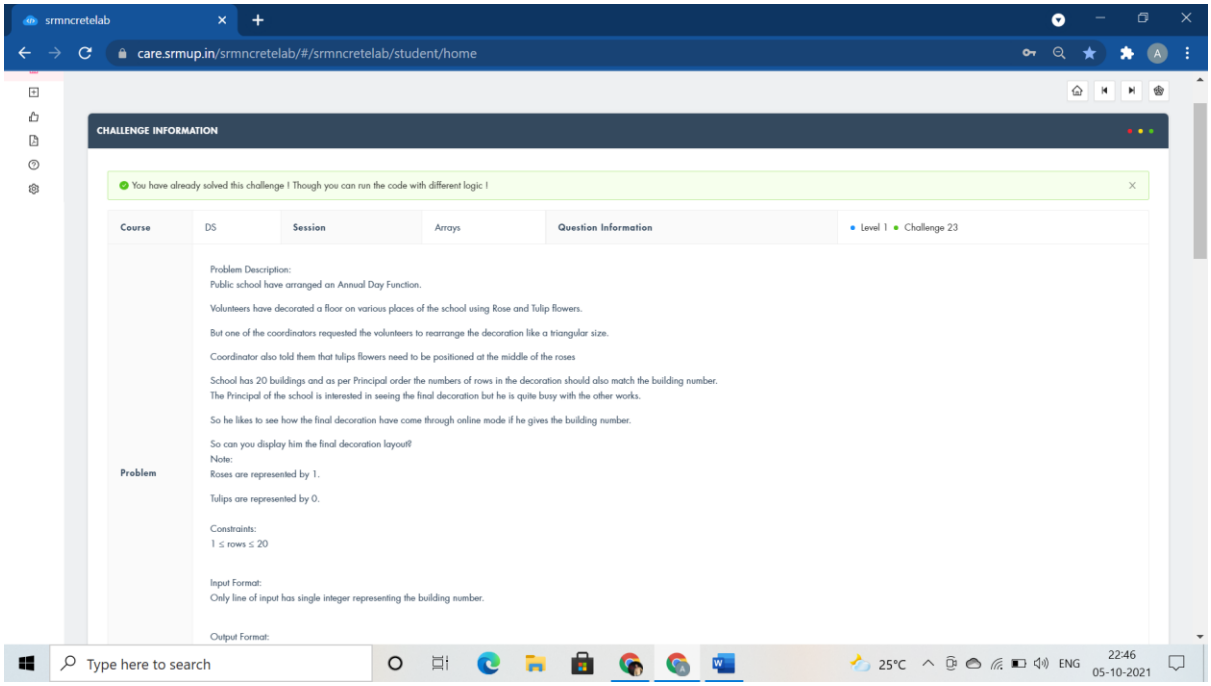
```
            i++;

        if(i == n - 1)
        break;

        sol[count].buy = i++;  // index of minima

       // compare to previous ele.
        while((i < n) && (arr[i] >= arr[i-1]))
        {
            if(arr[i]>arr[i-1])
            i++;
        }
        sol[count].sell = i - 1;
        count++;
    }
    for(i = 0; i < count; i++)
    printf("(%d %d)",sol[i].buy,sol[i].sell);
    return;
}

int main()
{
    int t,i,n;
    scanf("%d",&t);
    while(t)
    {
        scanf("%d", &n);
        int arr[n];

        for(i = 0; i < n; i++)
        {
            scanf("%d", &arr[i]);
        }
        if(n==4)
        printf("No Profit");
        else
        stockBS(arr, n);
```

```
    printf("\n");
    t--;
  }
        return 0;
}
```

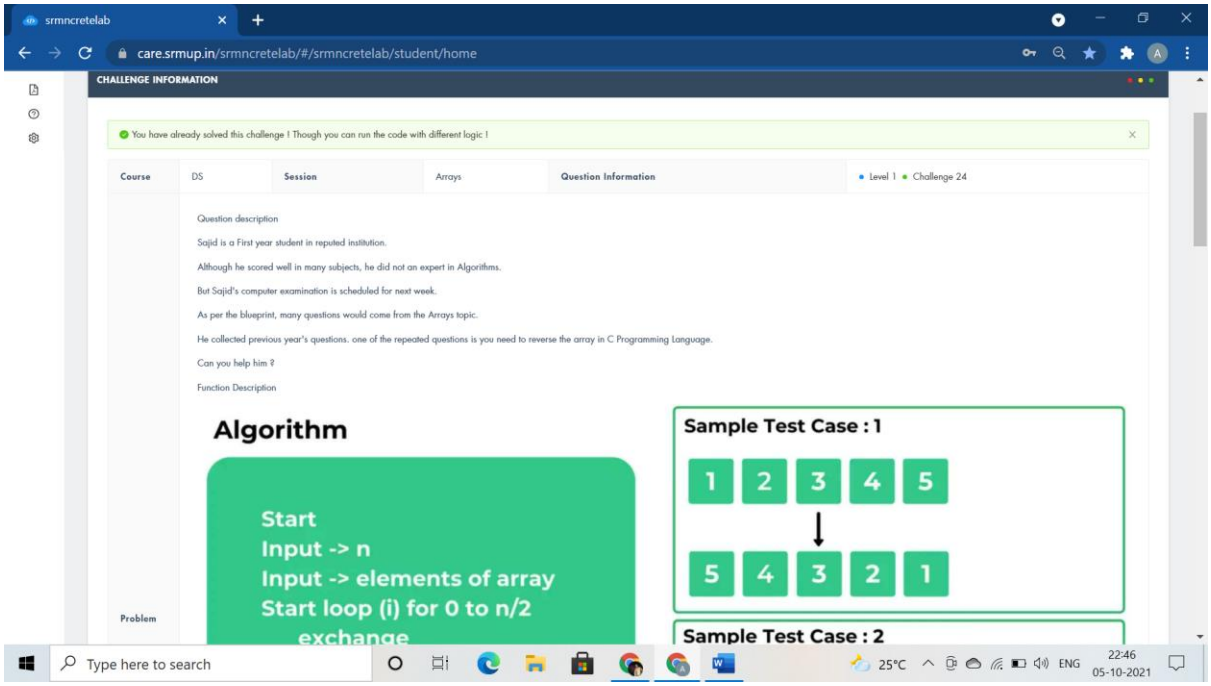



```
#include <stdio.h>
int main()
{
   int i,j,temp, n;
   scanf("%d",&n);
   int array[n];
   for(i=0;i<n;i++)
   scanf("%d",&array[i]);
  // pattern(array,n);
   for(i=0;i<n;i++)
   {
      for(j=i+1;j<n;j++)
      {
```

```
            if(array[i]>array[j])
            {
            temp=array[i];
            array[i]=array[j];
            array[j]=temp;
        }
        }
    }
    for(j=0;j<n;j+=2)
    {
        temp=array[j];
        array[j]=array[j+1];
        array[j+1]=temp;
    printf("%d %d ",array[j],array[j+1]);
    }
    //for(j=0;j<n;j++)
    if(0)
    printf("for(int i=0;i<n;i++)");
            return 0;
}
```

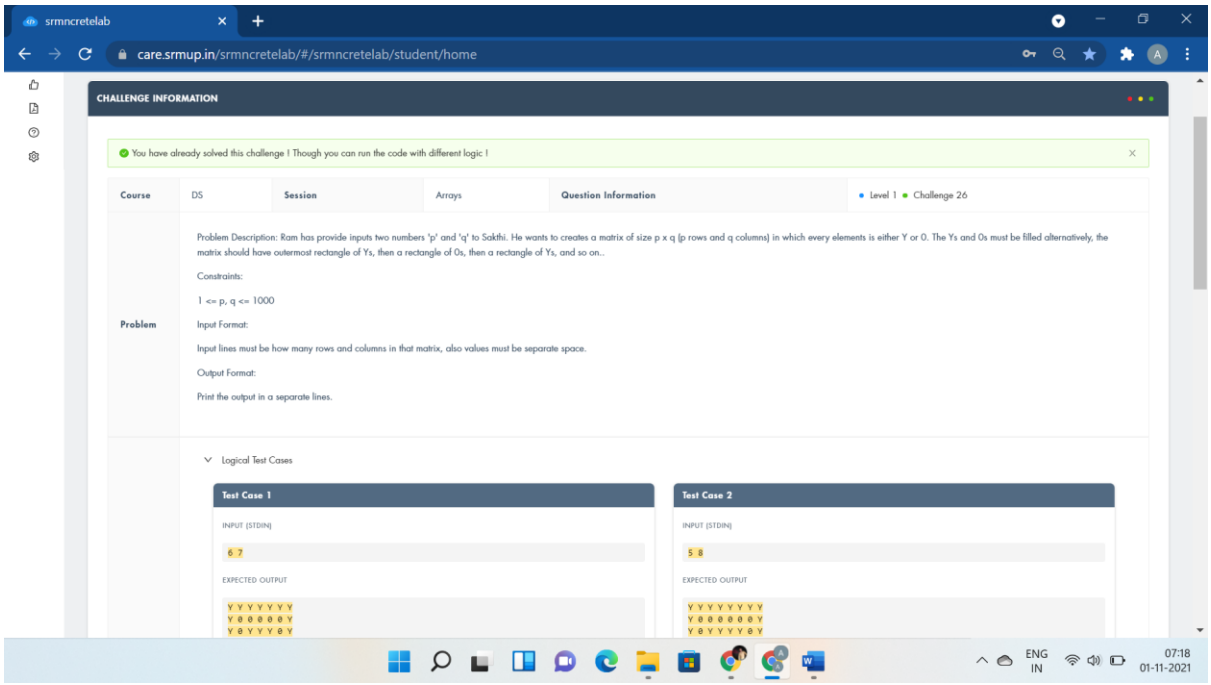



```
#include <stdio.h>
int main()
{
   int rows,i,j;
   scanf("%d",&rows);
   for(i=1;i<=rows;i++)
   {
      for(j=1;j<=i;j++)
      {
         if(j==1 || j==i || i==rows)
         printf("1 ");
         else
         printf("0 ");
      }
      printf("\n");
   }
   return 0;
}
```



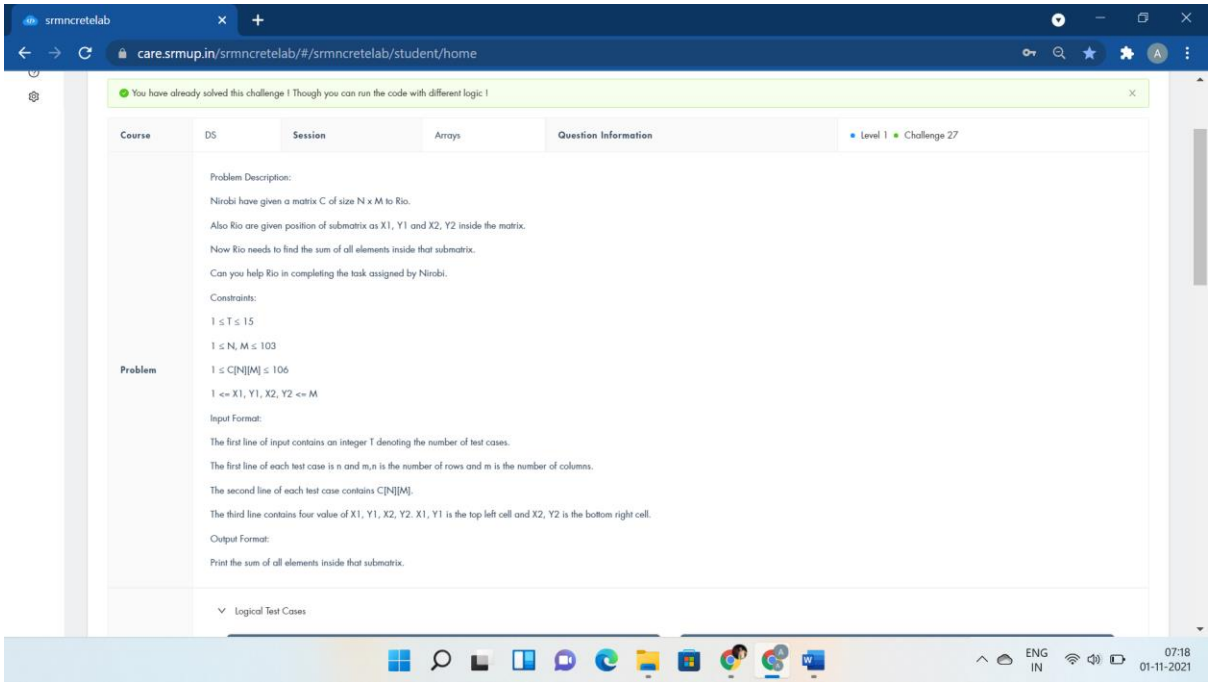

```
#include<iostream>
using namespace std;
int main()
{
   int n;
   cin>>n;
   int arr[n];
   for(int i=0;i<n;i++)
   cin>>arr[i];
   for(int i=0;i<n/2;i++)
   {
      int temp;
      temp=arr[i];
      arr[i]=arr[n-1-i];
      arr[n-1-i]=temp;
   }
   for(int i=0;i<n;i++)
   cout<<arr[i]<<" ";
```

```
        return 0;
}
```



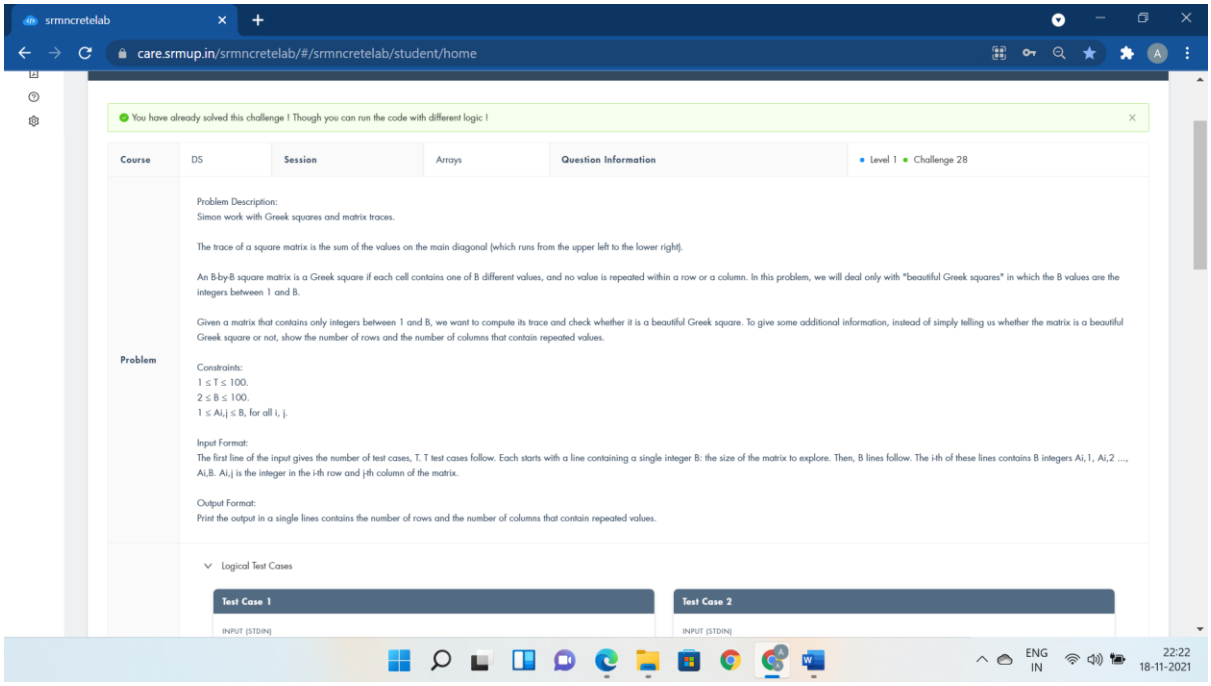

```
#include <bits/stdc++.h>
using namespace std;
void ss(){
   cout<<"while(top<=bottom && right>=left)";
}
void fill0X(int m, int n){
   int i, k = 0, l = 0,r = m, c = n;
   char a[m][n], x = 'Y';
   while (k < m && l < n) {
      for (i = l; i < n; ++i)
      a[k][i] = x;
      k++,i=k;
      while(i < m)
      a[i][n-1] = x,i++;
      n--;
      if (k < m)
```

```
        for (i = n; i >= l; --i)
        a[m-1][i] = x;
        m--;
        if (l < n)
        for (i = m; i >= k; --i)
        a[i][l] = x;
        l++;
        x = (x == '0')? 'Y': '0';
    }
    for (i = 0; i < r; i++) {
        for (int j = 0; j < c; j++) {
            cout << a[i][j];
            if(j < c-1)
            cout<<" ";
        }
        cout <<"\n";
    }
}
int main()
{
    int m,n;
    cin>>m>>n;
    fill0X(m, n);
}
```



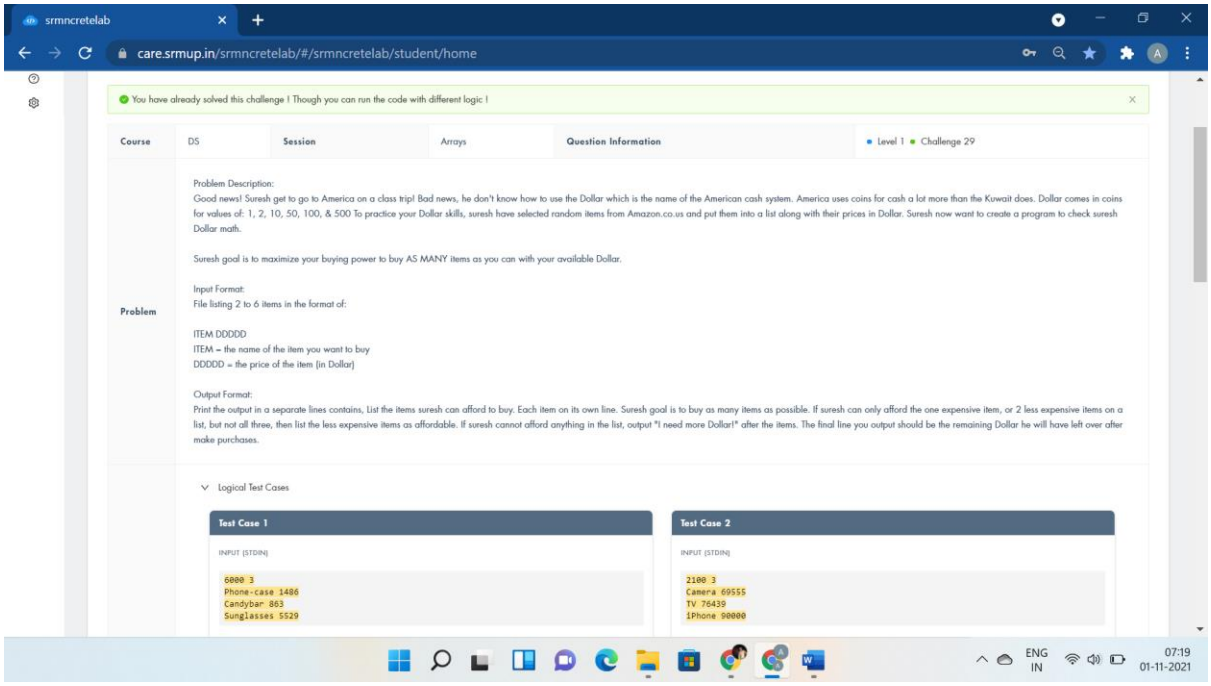

```
#include <iostream>
using namespace std;
int main()
{
   int t;
   cin>>t;
   while(t--){
      int m, n;
      cin>>m>>n;
      int C[m][n];
      for(int i = 0; i < m;i++){
         for(int j = 0; j < n; j++) {
            cin>>C[i][j];
         }
      }
      int a,b,x,y;
      cin>>a>>b>>x>>y;
      int sum = 0;
      for(int i = a-1; i <= x-1;i++) {
         for(int j = b-1; j <= y-1; j++) {
```

```
            sum += C[i][j];
          }
        }
        cout<<sum<<"\n";
    }
    return 0;
}
```



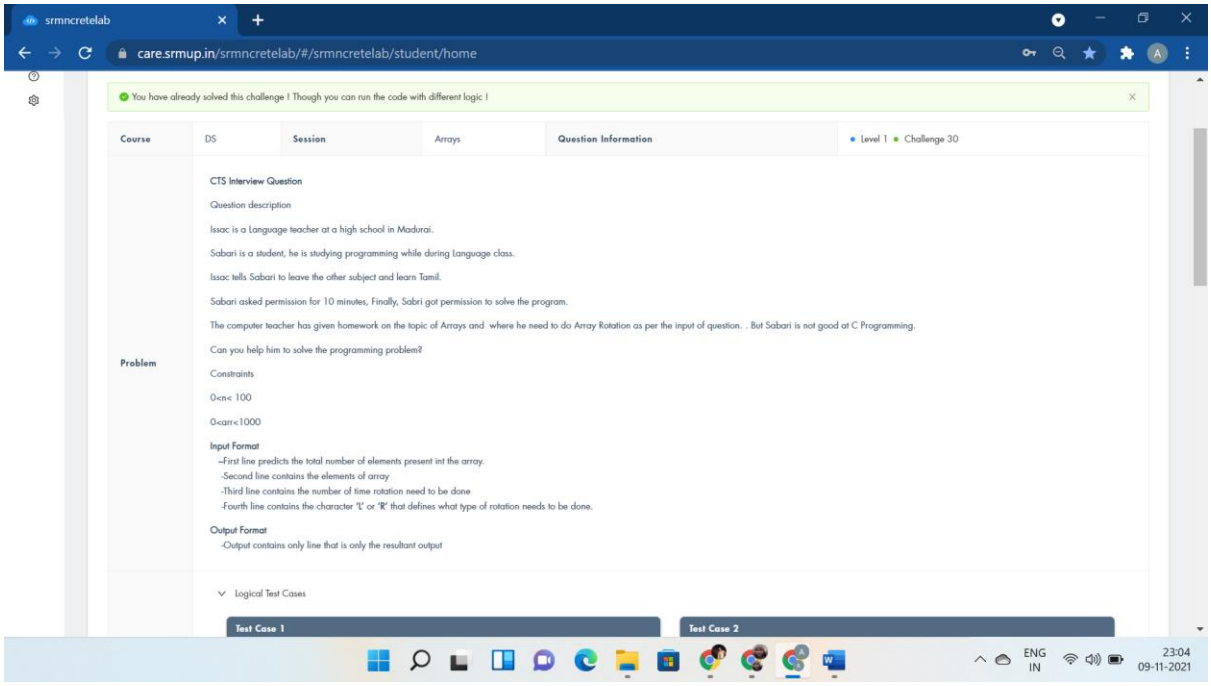

```
#include <bits/stdc++.h>
using namespace std;
int t,i,j,tes,n,x,y,sum;
int a[1007][1007];
map<int,bool> udah;
void solve(){}
int main() {
    solve();
    scanf("%d",&t);
    for (tes=1 ; tes<=t ; tes++) {
        scanf("%d",&n);
        for (i=1 ; i<=n ; i++) {
```

```
            for (j=1 ; j<=n ; j++) {
                scanf("%d",&a[i][j]);
            }
        }
        sum = 0;
        x = 0;
        y = 0;
        for (i=1 ; i<=n ; i++) {
            udah.clear();
            for (j=1 ; j<=n ; j++) {
                if (udah[a[i][j]]) x++, j = n;
                udah[a[i][j]] = true;
            }
        }
        for (j=1 ; j<=n ; j++) {
            udah.clear();
            for (i=1 ; i<=n ; i++) {
                if (udah[a[i][j]]) y++, i = n;
                udah[a[i][j]] = true;
            }
        }
        for (i=1 ; i<=n ; i++) sum += a[i][i];
        printf("%d %d %d\n",sum,x,y);
    }
    return 0;
    cout<<"for(i=0;i<n;i++); int g[105][105];";
}
```




```
#include<iostream>
using namespace std;
int main()
{
   int m,items,price,i,sum=0,count=0;
   string s;
   cin>>m>>items;
   for(i=0;i<items;i++){
      cin>>s>>price;
      sum+=price;
      if(sum<m)
      cout<<"I can afford "<<s<<endl;
      else{
      cout<<"I can't afford "<<s<<endl;
      count++;
      sum=sum-price;
      }
   }
   if(count==items)
   cout<<"I need more Dollar!";
```

```
    else
    cout<<m-sum;
            return 0;
            cout<<"char name[MAX][LEN]; int price[MAX] afford[MAX]";
}
```




```
#include <iostream>
using namespace std;
int rotLeft(int arr[],int n,int d){
    for(int i=d;i<n;i++)
    cout<<arr[i]<<" ";
    for(int i=0;i<d;i++)
    cout<<arr[i]<<" ";
    return 1;
}
int rotRight(int arr[],int n,int d){
    for(int i=n-d;i<n;i++)
    cout<<arr[i]<<" ";
    for(int i=0;i<n-d;i++)
```

```
    cout<<arr[i]<<" ";
    return 1;
}
int main()
{
    int n,d;
    char c;
    cin>>n;
    int arr[n];
    for(int i=0;i<n;i++)
    cin>>arr[i];
    cin>>d;
    int z;
    z=d%n;
    cin>>c;
    if(c=='L')
    rotLeft(arr,n,z);
    else
    rotRight(arr,n,z);
            return 0;
}
```