

October 22nd 2021, 8:53:59 pm



## CHALLENGE INFORMATION

✓ You have already solved this challenge ! Though you can run the code with different logic !

Course	DS	Session	Arrays	Question Information	<ul style="list-style-type: none"> <li>Level 1</li> <li>Challenge 21</li> </ul>
Problem	<p>Problem Description:</p> <p>Nirobi have given a matrix C of size N x M to Rio.</p> <p>Also Rio are given position of submatrix as X1, Y1 and X2, Y2 inside the matrix.</p> <p>Now Rio needs to find the sum of all elements inside that submatrix.</p> <p>Can you help Rio in completing the task assigned by Nirobi.</p> <p>Constraints:</p> <p><math>1 \leq T \leq 15</math></p> <p><math>1 \leq N, M \leq 103</math></p> <p><math>1 \leq C[N][M] \leq 106</math></p> <p><math>1 \leq X1, Y1, X2, Y2 \leq M</math></p>				

```
#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;
}

```

role student
name arian
ID 579410274420
dept School of Computing

Logout

er 22nd 2021, 8:54:39 pm

Home
Previous
Next
Flag

CHALLENGE INFORMATION

You have already solved this challenge ! Though you can run the code with different logic !

Course	DS	Session	Arrays	Question Information	<div> Level 1 Challenge 22 </div>
Problem	<p><b>Problem Description:</b> Simon work with Greek squares and matrix traces.</p> <p>The trace of a square matrix is the sum of the values on the main diagonal (which runs from the upper left to the lower right).</p> <p>An B-by-B square matrix is a Greek square if each cell contains one of B different values, and no value is repeated within a row or a column. In this problem, we will deal only with "beautiful Greek squares" in which the B values are the integers between 1 and B.</p> <p>Given a matrix that contains only integers between 1 and B, we want to compute its trace and check whether it is a beautiful Greek square. To give some additional information, instead of simply telling us whether the matrix is a beautiful Greek square or not, show the number of rows and the number of columns that contain repeated values.</p>				

```

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

```

```

#define lli long long int
void ss()
{
    return;
    cout<<" for(i=0;i<n;i++) void solve() int g[105][105];";
}
int main()
{

    int t;
    cin >> t;

    for (int test_case = 1; test_case <= t; test_case++) {
        int n;
        cin >> n;

        int** arr = (int**)(malloc(sizeof(int*) * n));
        lli sum = 0;
        int row = 0;

        for (int i = 0; i < n; i++) {
            arr[i] = (int*)(malloc(sizeof(int) * n));
            map<int, int> mymap;
            bool flag = false;
            for (int j = 0; j < n; j++) {
                cin >> arr[i][j];
                if (i == j)
                    sum += arr[i][j];
                if (flag == false && mymap.find(arr[i][j]) != mymap.end()) {
                    row++;
                    flag = true;
                }
                if (flag == false)
                    mymap[arr[i][j]]++;
            }
        }

        int col = 0;
        for (int i = 0; i < n; i++) {
            map<int, int> mymap;
            bool flag = false;
            for (int j = 0; j < n; j++) {
                if (flag == false && mymap.find(arr[j][i]) != mymap.end()) {
                    col++;
                    flag = true;
                    break;
                }
            }
            if (flag == false)

```

```

        mymap[arr[j][i]]++;
    }
}
cout<< sum << " " << row << " " << col << endl;
}

return 0;
}

```

October 22nd 2021, 8:55:31 pm
🏠 ⏮ ⏭ 🏠

### CHALLENGE INFORMATION

✔ You have already solved this challenge ! Though you can run the code with different logic !

Course	DS	Session	Arrays	Question Information	<span>● Level 1</span> <span>● Challenge 23</span>
<b>Problem</b>	<p><b>Problem Description:</b> Public school have arranged an Annual Day Function.</p> <p>Volunteers have decorated a floor on various places of the school using Rose and Tulip flowers.</p> <p>But one of the coordinators requested the volunteers to rearrange the decoration like a triangular size.</p> <p>Coordinator also told them that tulips flowers need to be positioned at the middle of the roses</p> <p>School has 20 buildings and as per Principal order the numbers of rows in the decoration should also match the building number.</p> <p>The Principal of the school is interested in seeing the final decoration but he is quite busy with the other works.</p> <p>So he likes to see how the final decoration have come through online mode if he gives the building number.</p> <p>So can you display him the final decoration layout?</p> <p><b>Note:</b></p>				

```

#include <iostream>
using namespace std;
int main()
{
    int rows;
    cin>>rows;
    int i;
    for(i=1;i<=rows;i++)
    {
        for(int j = 1; j <= i; j++)
        {
            if(j == i || j == 1 || i == rows)  cout<<"1 ";

```

```

        else cout<<"0 ";
    }
    cout<<"\n";
}
return 0;
}

```

role

student

name

aryan

ID

579410274420

dept

school of computing

Logout

October 22nd 2021, 8:56:08 pm

Home

Back

Next

Profile

CHALLENGE INFORMATION

✔

You have already solved this challenge ! Though you can run the code with different logic !

×

Course	DS	Session	Arrays	Question Information	<div> <div>Level 1</div> <div>Challenge 24</div> </div>
				<p>Question description</p> <p>Simon is studying B.Tech.-Mechanical Engineering.</p> <p>He's going to attend a computer science-based subject exam this semester.</p> <p>Due to the less preparation in the previous monthly tests, his internal mark decreased.</p> <p>His computer science Professor made an offer one more chance to boost up his internal marks.</p> <p>Professor assigns a program to Simon for the internal mark boostup.</p> <p>So Simon wants to identify the element of array which occurs most time in the array</p> <p>Can you help him ?</p> <p>Function Description</p>	

1

2

1

2

1

4

5

1

2

```

#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin>>n;
    int arr[n];

    for(int i=0;i<n;i++)
    {
        cin>>arr[i];
    }
}

```

```

int i;
unordered_map<int, int> hash;
for(i= 0;i< n;i++)
    hash[arr[i]]++;
int max_count = 0, res = -1;
for (auto i : hash) {
    if (max_count < i.second) {
        res = i.first;
        max_count = i.second;
    }
}

cout<<res<<"\n";

return 0;
}

```

October 22nd 2021, 8:56:37 pm

CHALLENGE INFORMATION

✔ You have already solved this challenge ! Though you can run the code with different logic !

Course	DS	Session	Arrays	Question Information
				<div> Level 1 </div> <div> Challenge 25 </div>

Question description

Malar is a First year student in reputed institution.  
Although he scored well in many subjects, he did not an expert in Algorithms.  
But malar's computer examination is scheduled for next week.  
As per the blueprint, many questions would come from the Arrays topic.  
He collected previous year's questions. one of the repeated questions is you need to find the pairs in Array with given sum.  
Can you help him ?

Function Description

## Sample Test Case

```

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

int main()

```

```

{

int n;
cin>>n;
int array[n];
int i;
for(i=0;i<n;i++)
    cin>>array[i];
int num,j,count=0;
cin>>num;
vector<int>v ;
for(int i =0; i< n; i++)
{
    for(j=i+1;j<n;j++)
    {
        if(array[i] +array[j] == num)
        {
            cout<<"["<<array[i]<<" "<<array[j]<<"]\n";
            count++;
        }
    }
}
}
cout<<"Total Number of Pairs:"<<count;
return 0;
}

```

role

student

name

aryan

ID

375720274726

dept

school of computing

Logout

ber 22nd 2021, 8:57:28 pm

🏠

⏮

⏭

🏠

CHALLENGE INFORMATION

🟢 You have already solved this challenge ! Though you can run the code with different logic !

Course	DS	Session	Arrays	Question Information	<div><div>●</div> Level 1</div> <div><div>●</div> Challenge 26</div>
Problem	<div>Problem Description:</div> <div>saravanan with his friends going to the theatre for a movie.</div> <div>The seating arrangement is triangular in size.</div> <div>Theatre staffs insisted the audience to sit in odd row if the seat number is odd and in even row if the seat number is even.</div> <div>But the instruction is very confusing for saravanan and his friends.</div> <div>So help them with the seating layout so that they can sit in correct seats.</div> <div>Constraints:</div> <div><math>4 \leq N \leq 20</math></div> <div>Input Format:</div> <div>Only line of input has single integer value representing the number of rows in the theatre</div>				

```
#include <iostream>
using namespace std;
int main()
{

int N;
cin>>N;
int i;
for(i=1;i<=N;i++)
{
    int s = 1,p=2,j=0;
    while(j < i*2)
    {
        if(i % 2 == 1)
            cout<<j+s<<" ";
        else
            cout<<j+p<<" ";
    }
}
```



```

        j+=2;
    }
    // j+=2;
    cout<<"\n";
}

}

```

ROLE student Name aayan ID 375410274420 Dept School of Computing

er 22nd 2021, 8:58:10 pm

## CHALLENGE INFORMATION

You have already solved this challenge ! Though you can run the code with different logic !

course	DS	Session	Arrays	Question Information	<ul style="list-style-type: none"> <li>Level 1</li> <li>Challenge 27</li> </ul>
problem	<p><b>Problem Description:</b></p> <p>Good news! Suresh get to go to America on a class trip! Bad news, he don't know how to use the Dollar which is the name of the American cash system. America uses coins for cash a lot more than the Kuwait does. Dollar comes in coins for values of: 1, 2, 10, 50, 100, &amp; 500 To practice your Dollar skills, suresh have selected random items from Amazon.co.us and put them into a list along with their prices in Dollar. Suresh now want to create a program to check suresh Dollar math.</p> <p>Suresh goal is to maximize your buying power to buy AS MANY items as you can with your available Dollar.</p> <p><b>Input Format:</b></p> <p>File listing 2 to 6 items in the format of:</p> <p>ITEM DDDDD</p> <p>ITEM = the name of the item you want to buy</p> <p>DDDDD = the price of the item (in Dollar)</p>				

```

#include <bits/stdc++.h>
using namespace std;
int main()
{
    int MAX,LEN;

```

```

cin>>MAX>>LEN;
string a[LEN];
int b[LEN],c[LEN],count=0,i,j;
for(i=0;i<LEN;i++){cin>>a[i]>>b[i];c[i]=b[i];}
for(i=0;i<LEN;i++){
    if(MAX-c[i]>0){
        for(j=0;j<LEN;j++){cout<<"hello there";
            if(c[i]==b[j]){cout<<"hello there";
                cout<<"I can afford "<<a[i]<<endl;
                MAX-=c[i];
                break;
            }
        }
    }
    else{
        for(j=0;j<LEN;j++){
            if(c[i]==b[j]){cout<<"hello there";
                cout<<"I can't afford "<<a[i]<<endl;
                count++;
                break;
            }
        }
    }
}
(count==LEN) ? cout<<"I need more Dollar!" : cout<<MAX;
return 0;
cout<<"char name[MAX][LEN]; int price[MAX] afford[MAX] for(i=0;i<items;i++)";
}

```

role student
name aryan
ID 579410274420
dept School of Computing

Logout

er 22nd 2021, 8:59:10 pm

Home
Previous
Next
Profile

CHALLENGE INFORMATION

You have already solved this challenge ! Though you can run the code with different logic !

Course	DS	Session	Arrays	Question Information	<div>Level 1</div> <div>Challenge 28</div>
<div>TCS Interview Question</div> <div>Question description</div> <p>Ravi participated in TCS off campus interview at reputed institution, one of the technical question he has to complete with in the given time, where you need to sort the array in the waveform. There might be multiple possible output of the program. the following pattern output is appreciated.</p> <div>Function Description</div> <p>This is a simple method of solving this question which contains basic 2 steps and they are as follow</p> <p><b>Step : 1</b> – Sort the array in ascending order.</p> <p><b>Step : 2</b> – Swap all adjacent elements of the array</p> <p>Let us consider the input array be {3, 6, 5, 10, 7, 20}. After sorting, we get {3, 5, 6, 7, 10, 20}. After swapping adjacent elements, we get {5, 3, 7, 6, 20, 10}.</p> <div> <div>Brute Force Method : Step 1</div> <div></div> <div>Brute Force Method : Step 2</div> </div>					

```

#include <bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin>>n;
    int array[n];
    for(int i= 0; i < n ; i++)
    {
        cin>>array[i];
    }

    sort(array, array + n);
    int i;
    for(i=0;i<n;i++)
    {
        swap(array[i], array[i+1]);
        i++;
    }
}

```

```

}
for(int s : array)
cout<<s<<" ";

    return 0;
    cout<<"if(array[i]>array[j])";
}

```

role Student
name ayan
id 975410274420
dept school of computing

Logout

October 22nd 2021, 8:59:36 pm

Home
Previous
Next
Flag

CHALLENGE INFORMATION

You have already solved this challenge ! Though you can run the code with different logic !

Course	DS	Session	Arrays	Question Information	<div>Level 1</div> <div>Challenge 29</div>
Problem	<p><b>Problem Description:</b></p> <p>Umesh has <math>n</math> mixtures in front of him, arranged in a row. Each mixture has one of 100 different colors (colors have numbers from 0 to 99).</p> <p>He wants to mix all these mixtures together. At each step, he is going to take two mixtures that stand next to each other and mix them together, and put the resulting mixture in their place.</p> <p><b>Functional Description:</b></p> <p>When mixing two mixtures of colors <math>a</math> and <math>b</math>, the resulting mixture will have the color <math>(a+b) \bmod 100</math>.</p> <p>Also, there will be some smoke in the process. The amount of smoke generated when mixing two mixtures of colors <math>a</math> and <math>b</math> is <math>a*b</math>.</p> <p><b>Find out what is the minimum amount of smoke that Umesh can get when mixing all the mixtures together.</b></p> <p><b>Constraints:</b></p>				

```

#include<bits/stdc++.h>
using namespace std;
int main()
{
    int n;
    cin>>n;
    while(n!=-1)
    { long long a[n];

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

```

```

{
    cin>>a[i];
}

vector<vector<long long>>dp(n,vector<long long>(n+1,1000000000000000000));
long long sum[n][n+1];
for(int i=0;i<n;i++)
{
    sum[i][1]=a[i];
    dp[i][1]=0;
}

for(int len=2;len<=n;len++)
{
    for(int i=0;i<=n-len;i++)
    {
        for(int j=1;j<len;j++)
        {
            sum[i][len]=(sum[i][j]+sum[i+j][len-j])%100;
            long long x=dp[i][j]+dp[i+j][len-j]+(sum[i][j]*sum[i+j][len-j]);
            dp[i][len]=min(x,dp[i][len]);
        }
    }
}
cout<<dp[0][n]<<endl;
n=-1;cin>>n;
}
return 0;
cout<<"scount[100][100] colours[100]";
}

```

role

student

name

aryan

ID

379410274420

dept

School of Computing

Logout

October 22nd 2021, 9:00:17 pm

Home

Previous

Next

Profile

CHALLENGE INFORMATION

✔ You have already solved this challenge ! Though you can run the code with different logic !

×

Course	DS	Session	Arrays	Question Information	<div> <div>Level 1</div> <div>Challenge 30</div> </div>
Problem	<p>Problem Description: Ram has provide inputs two numbers 'p' and 'q' to Sakthi. He wants to creates a matrix of size p x q (p rows and q columns) in which every elements is either Y or O. The Ys and Os must be filled alternatively, the matrix should have outermost rectangle of Ys, then a rectangle of Os, then a rectangle of Ys, and so on..</p> <p>Constraints:</p> <p>1 &lt;= p, q &lt;= 1000</p> <p>Input Format:</p> <p>Input lines must be how many rows and columns in that matrix, also values must be separate space.</p> <p>Output Format:</p> <p>Print the output in a separate lines.</p>				

```
#include <bits/stdc++.h>
using namespace std;
void ss()
{
    cout<<"while(top<=bottom && right>=left)";
}
void fillOX(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;
    fillOX(m, n);
}

```