

(1)

```
#include<iostream>
#include<string>
using namespace std;

struct Restaurant
{
    int Id;
    string Name;
    int Ratings[10];
    float AverageRating;
};

void input(Restaurant restaurants[], int size);
void output(Restaurant restaurants[], int size);

void main()
{
    int size;
    cout << "Enter number of restaurants: ";
    cin >> size;

    Restaurant restaurants[25];
    input(restaurants, size);
    output(restaurants, size);
}

void input(Restaurant restaurants[], int size)
{
    int ratingsSize;
    for (int i = 0; i < size; i++)
    {
        cout << "Enter Id: ";
        cin >> restaurants[i].Id;

        cout << "Enter Name: ";
        cin >> restaurants[i].Name;

        restaurants[i].AverageRating = 0;
        cout << "Enter number of ratings: ";
        cin >> ratingsSize;

        cout << "Enter Ratings: ";
        for (int j = 0; j < ratingsSize; j++)
        {
            cin >> restaurants[i].Ratings[j];
            restaurants[i].AverageRating += restaurants[i].Ratings[j];
        }
        restaurants[i].AverageRating = restaurants[i].AverageRating /
ratingsSize;

        cout << "=====" << endl;
    }
}

void output(Restaurant restaurants[], int size)
{

```

```
cout << "Output: " << endl;
for (int i = 0; i < size; i++)
{
    cout << "Restaurant Id: " << restaurants[i].Id
        << "    Restaurant Name: " << restaurants[i].Name
        << "    Restaurant Average Rating: " <<
restaurants[i].AverageRating << endl;
}
}
```

(2)

```
#include <iostream>
#include<string>
using namespace std;
struct player
{
    string name;
    int scores[3];
    int total;
};
int main()
{
    int num;
    cout<<"enter number of players"<<endl;
    cin>>num;
    player *arr=new player[num];
    for(int i=0;i<num;i++)
    {
        cout<<"player #"<<i+1<<endl;
        cout<<"Name: ";
        cin>>arr[i].name;
        cout<<"Scores: ";
        arr[i].total=0;
        for(int y=0;y<3;y++)
        {
            cin>>arr[i].scores[y];
            arr[i].total+=arr[i].scores[y];
        }

    }
    int max=-100;
    int max_pos;
    for(int k=0;k<num;k++)
    {
        if(arr[k].total>max)
        {
            max=arr[k].total;
            max_pos=k;
        }
    }
    cout<<"the player with maximum score is "<<arr[max_pos].name<<endl;
    return 0;
}
```

(3)

```
#include <iostream>
using namespace std;

void input(int arr[][10], int rows, int cols);
void output(int arr[][10], int rows, int cols);

int main()
{
    int rows, cols;
    int arr[10][10];

    cout << "Enter number of rows: ";
    cin >> rows;

    cout << "Enter number of cols: ";
    cin >> cols;

    input(arr, rows, cols);

    int rowIndex0, rowIndex1;
    cout << "Enter first row index: ";
    cin >> rowIndex0;

    cout << "Enter second row index: ";
    cin >> rowIndex1;

    for (int j = 0; j < cols; j++)
    {
        swap(arr[rowIndex0][j], arr[rowIndex1][j]);
    }

    output(arr, rows, cols);

    system("Pause");
    return 0;
}

void input(int arr[][10], int rows, int cols)
{
    cout << "Enter array: " << endl;
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
            cin >> arr[i][j];
        }
    }
}

void output(int arr[][10], int rows, int cols)
{
    cout << "Output array: " << endl;
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < cols; j++)
        {
```

```
        cout<< arr[i][j] << " ";
    }
    cout << endl;
}
```

(4)

```
#include<iostream>
using namespace std;
int GetElementLocation(int m[][10], int rows, int cols, int num1, int num2);
void main()
{
    int rows, cols;
    int matrix[10][10];
    cout << "Enter matrix size ( rows , cols ) : ";
    cin >> rows >> cols;

    cout << "Enter the matrix : ";
    for (int i = 0; i < rows; i++)
        for (int j = 0; j < cols; j++)
            cin >> matrix[i][j];

    int num1, num2;
    cout << "Enter two existing numbers : ";
    cin >> num1 >> num2;

    int distance = GetElementLocation(matrix, rows, cols, num1, num2);
    cout << "The distance between them is : " << distance;

}
```

```
int GetElementLocation(int m[][10], int rows, int cols, int num1, int num2)
{
    int element1Row, element2Row, element1Col, element2Col;
    for (int i = 0; i < rows; i++)
        for (int j = 0; j < cols; j++)
            if (m[i][j] == num1)
            {
                element1Row = i;
                element1Col = j;
            }
            else if (m[i][j] == num2)
            {
                element2Row = i;
                element2Col = j;
            }
}
```

```
    return abs(element1Row - element2Row) + abs(element1Col - element2Col);  
}
```

(5)

```
#include<iostream>
```

```
using namespace std;
```

```
void input(int m[][10], int &r, int &c);
```

```
void getMaxMinandAvg(int m[][10], int rows, int cols, int row, int& Max, int& Min, int& Avg);
```

```
void main()
```

```
{
```

```
    int matrix[10][10], rows, cols, r;
```

```
    int Max, mini, Average;
```

```
    input(matrix, rows, cols);
```

```
    cout << "Select Row\n";
```

```
    cin >> r;
```

```
    getMaxMinandAvg(matrix, rows, cols, r, Max, mini, Average);
```

```
    cout << "Min: " << mini << endl << "Max: " << Max << endl << "Average : " << Average  
<< endl;
```

```
}
```

```
void input(int m[][10], int& r, int& c)
```

```
{
```

```
    cout << "Enter the size of the 2D array:\n";
```

```
    cin >> r >> c;
```

```
    cout << "Enter the array :\n";
```

```
    for (int i = 0; i < r; i++)
```

```
        for (int j = 0; j < c; j++)
```

```
            cin >> m[i][j];
```

```
}
```

```
void getMaxMinandAvg(int m[][10], int rows, int cols, int row, int& Max, int& Min, int& Avg)
```

```
{
```

```
    row--;
```

```
    Max=Min = m[row][0];
```

```
    Avg = 0;
```

```
    for (int j = 0; j < cols;j++)
```

```
    {
```

```
        Avg += m[row][j];
```

```
        if (m[row][j]>Max)
```

```
            Max = m[row][j];
```

```
        if (m[row][j] < Min)
```



```
                Min = m[row][j];
            }
            Avg = Avg / cols;
        }
```

(6)

```
#include <iostream>
#include<string>
using namespace std;
void insertsorted(int number,int arr[],int size,int num_elemnt)
{
    bool flag=false;
    int pos=0;
    for(int i=0;i<size;i++)
    {
        if(number<arr[i])
        {
            pos=i;
            flag=true;
            break;
        }
    }

    if(flag==true)
    {
        for(int o=num_elemnt-2;o>=pos;o--)
        {
            arr[o+1]=arr[o];
        }
        arr[pos]=number;
        cout<<"List after insertion:"<<endl;
        for(int y=0;y<=num_elemnt;y++)
        {
            cout<<arr[y]<<" , ";
        }
    }
}

int main()
{
    int prtarr[10];
    cout<<"Enter sorted list"<<endl;
    int x;
    int c=0;
    for(int i=0;i<10;i++)
    {
        cin>>x;
        if(x!=-1)
        {
            prtarr[i]=x;
            c++;
        }
        else
            break;
    }
    int num;
```

```
    cout<<"enter number"<<endl;
    cin>>num;
    insertsorted(num,ptrarr,10,c);

    return 0;
}
```

```

(7)#include<iostream>
#include<string>
using namespace std;

struct product
{
    int id;
    float price;
    string Name;
};

void input(product p[], int n);
void Replace(product p[], int n, string oldName, string NewName);
void display(product p[], int n);

void main()
{
    product *arr;
    int size;
    string oldname, newname;
    cout << "Enter the number of products:\n";
    cin >> size;
    arr = new product[size];
    input(arr, size);
    cout << "Enter the name to change: \n";
    cin >> oldname;
    cout << "Enter the new name:\n";
    cin >> newname;
    Replace(arr, size, oldname, newname);
    display(arr, size);
}

void input(product p[], int n)
{
    for (int i = 0; i < n; i++)
    {
        cout << "Enter the ID : \n";
        cin >> p[i].id;
        cout << "Enter the name : \n";
        cin >> p[i].Name;
        cout << "Enter the price: \n";
        cin >> p[i].price;
    }
}

void Replace(product p[], int n, string oldName, string NewName)
{
    for (int i = 0; i < n; i++)
    {
        if (p[i].Name == oldName)
            p[i].Name = NewName;
    }
}

```

```
}  
void display(product p[], int n)  
{  
    for (int i = 0; i < n; i++)  
    {  
        cout << "ID : "<< p[i].id<<" the Name : "<< p[i].Name<<" the Price: "<<  
p[i].price<<endl;  
    }  
}
```

(8)

```
#include <iostream>
using namespace std;
struct hotel [1.5 marks]
{
    char name[30];
    float rating ;
    float pricePerNight;
};
void input( hotel arr[] ,int size) [1.5 marks]{

    for(int i=0; i<size; i++) {
        cin>>arr[i].name;
        cin>>arr[i].rating;
        cin>>arr[i].pricePerNight;
    }
}
int displayhotels(hotel arr[], int size, int pay) [4 marks] {
    float maxrating =0; //0.5
    int maxindex=0;
    for(int i=0; i<size; i++)//0.5
    {
        if(arr[i].pricePerNight<=pay) // 0.5
        {

            cout<<arr[i].name<<" "<<arr[i].pricePerNight<<endl; // 0.5
            if(maxrating<arr[i].rating) // 0.5
            {
                maxrating=arr[i].rating; //0.5
                maxindex=i; //0.5
            }
        }
    }
    return maxindex; //0.5
}

int main() [3 marks]
{
    int size; // 0.5 all declarations
    cin>>size; //0.5 both cin
    hotel arr [10];
    input(arr,size); //0.5
    float pay;
```

```
cin>>pay;
int max = displayhotels(arr,size,pay); //0.5
cout<<"the best hotel is "<< arr[max].name<<"with rating"<<arr[max].rating; // 1
}
```