Array related problems

Write a program to read ‘n’ numbers from the keyboard (where ‘n’ is defined by the programmer) to store it in an one-dimensional array and to display the content of that array onto the video screen.

```
#include <iostream>
using namespace std;
int main()
{
    int a[100];
    int i,n;
    cout <<" How many numbers are in the array ? \n";
    cin >> n;
    cout <<"Enter the elements \n";
    for (i = 0; i <= n-1; ++i)
        cin >> a[i];
    cout <<"Contents of the array \n";
    for (i = 0; i <= n-1; ++i)
        cout << a[i] << '\t';
    return 0;
}
```

A program to read a set of numbers from the keyboard and to find out the sum of all elements of the given array using a function.

```
#include <iostream>

using namespace std;

int main()

{

    int a[100];




    int i,n,larg;

    cout <<" How many numbers are in the array ? \n";

    cin >> n;

    cout <<"Enter the elements \n";

    for (i = 0; i <= n-1; ++i)

        cin >> a[i];

    cout <<"Contents of the array \n";

    for (i = 0; i <= n-1; ++i)

        cout << a[i] << '\t';

    larg = a[0];

    for ( i = 0; i <= n-1; ++i)

    {

        if ( larg < a[i])

            larg = a[i];

    }

    cout <<"\n Largest value in the array = " << larg;

    return 0;

}
```

A program to read a set of numbers from the keyboard and to sort out the given array of elements in ascending order using a function.

```
#include<iostream>
using namespace std;
void read(int [],int);
void sort (int [],int);
void swap(int[],int[]);
void display(int [],int);
int main()
{

    int size;
    cout << "Enter the size of the array: ";
    cin >> size;
    int arr[size];
    read(arr,size);
    cout<<"Unsorted array"<<endl;
    cout<<"==============="<<endl;
    display(arr,size);
    sort(arr,size);
    cout<<"Sorted array"<<endl;
    cout<<"==============="<<endl;
    display(arr,size);
}
void read(int arr[],int size)
{

    for (int i = 0; i < size; ++i)
    {
        cout<<"a["<<i<<" ]=";
        cin>>arr[i];
    }
```

```
}
void display(int a[],int n)
{

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

        cout<<"a["<<i<<"]=";
        cout<<a[i];
        cout<<endl;
    }
}
void sort (int a[],int n)
{

    for (int i = 0; i <= n-1; ++i)
    {
        for (int j = 0; j <= n-1; ++j)
            if(a[i]<a[j])
                swap(a[i],a[j]);
    }
}
void swap(int a,int b)
{
    int temp;
    {
        temp = a;
        a = b;
        b = temp;
    }
}
```

A program to read a set of numbers from the keyboard and to sum out the given array of elements in using a function.

```
#include <iostream>
using namespace std;
const int MAX = 100;
void display (int [],int );
int sumarray(int [],int );
int main()
{
// function declaration  int sumarray(int a[], int n);
    int a[MAX];
    int i,n,sum;

    cout <<"How many numbers are in the array ? \n";
    cin >> n;
    cout <<"Enter the elements \n";
    for (i = 0; i <= n-1; ++i)
        cin >> a[i];
    cout <<"contents of the array \n";
    display (a,n);
    sum = sumarray(a,n);
    cout <<"\n sum of the elements of the array = " << sum;
    cout <<"\n";
    return 0;
}
void display (int a[],int n)
{
    int i;
    for (i = 0; i <= n-1; ++i)
        cout << a[i] << '\t';
}
int sumarray(int x[],int max)
```

```
{
    int i,temp = 0;
    for (i = 0; i <= max-1; ++i)
        temp = temp +x[i];
    return(temp);
}
```

A program to read a set of numbers and store it as a one-dimensional array again read a number n and check whether it is present in the array. If it is so, print the position of n in the array and also check whether it is repeated in the array.

```
#include <iostream>
using namespace std;
void searchNumber(int [], int , int );

int main()
{
    int size;
    cout << "Enter the size of the array: ";
    cin >> size;
    int arr[size];
    cout << "Enter " << size << " numbers:" << endl;
    for (int i = 0; i < size; ++i)
    {
        cin >> arr[i];
    }
    int searchNum;
    cout << "Enter the number to search: ";
    cin >> searchNum;
    searchNumber(arr, size, searchNum);
    return 0;
}
void searchNumber(int arr[], int size, int n)
```

```
{
    int found = 0;
    int repeated = 0;
    for (int i = 0; i < size; ++i)
    {
        if (arr[i] == n)
        {
            found = 1;
            cout << "Number " << n << " is present at position " << i << " in
the array." << endl;
            if (repeated==0)
            {
                repeated = 1;
                // Check if the number is repeated
                for (int j = i + 1; j < size; ++j)
                {
                    if (arr[j] == n)
                    {
                        cout << "Number " << n << " is repeated in the
array." << endl;
                        break;
                    }
                }
            }
        }
    }
    if (found==0)
    {
        cout << "Number " << n << " is not present in the array." << endl;
    }
}
```

Write and test the following function that returns through its reference parameters both the largest and the second largest values (possibly equal) stored in an array:

```
void largest(float& max1, float& max2, float a[], int n);
```

```
#include <iostream>
using namespace std;
const int MAX = 100;
void largest(float&, float&, float [], int );

int main()
{
    float a[MAX];
    int i,n;
    float max1,max2;

    cout <<"How many numbers are in the array ? \n";
    cin >> n;
    cout <<"Enter the elements \n";
    for (i = 0; i <= n-1; ++i)
        cin >> a[i];
    largest(max1,max2,a,n);
    cout<<"The 1st maximum value is "<<max1<<endl;
    cout<<"The 2nd maximum value is "<<max2<<endl;

}

void largest(float& max1, float& max2, float a[], int n)
{

    int i1=0, i2;
    for (int i=1; i<n; i++)
    {
```

```
        if (a[i] > a[i1])
            i1 = i;
    }
     max1 = a[i1];

    i2 = ( i1 == 0 ? 1 : 0 );
    for (int i=i2+1; i<n; i++)
    {

        if (i != i1 && a[i] > a[i2])
            i2 = i;
    }
    max2 = a[i2];
}
```

Write and test the following function that returns through its reference parameters both the maximum and the minimum values stored in an array:
`void getExtremes(float& min, float& max, float a[], int n);`

```
#include <iostream>
using namespace std;
const int MAX = 100;
void getExtremes(float&, float&, float [], int );

int main()
{
    float a[MAX];
    int i,n;
    float min,max;

    cout <<"How many numbers are in the array ? \n";
    cin >> n;
    cout <<"Enter the elements \n";
```

```
    for (i = 0; i <= n-1; ++i)
        cin >> a[i];
    getExtremes(min,max,a,n);
    cout<<"The  maximum value is "<<max<<endl;
    cout<<"The  minimum value is "<<min<<endl;
}
void getExtremes(float& min, float& max, float a[], int n)
{
    min = max = a[0];
    for (int i=1; i<n; i++)
        if (a[i] < min)
            min = a[i];
        else if (a[i] > max)
        max = a[i];
}
```

Write and test the function `int frequency(float a[], int n, int x);` This function counts the number of times the item x appears among the first n elements of the array a and returns that count as the frequency of x in a.

```
#include <iostream>
#include<cstdlib>
#include<ctime>
using namespace std;
void read(int[],int);
int frequency(int [], int, int );
void display(int[],int);
int main()
{
    srand(time(0));
    int size;
    cout << "Enter the size of the array: ";
    cin >> size;
```

```
    int arr[size];
    read(arr,size);
    display(arr,size);
    int searchNum;
    cout << "Enter the number to search: ";
    cin >> searchNum;
    cout << searchNum << " has frequency " << frequency(arr,size,searchNum)
<< endl;
    return 0;
}
int frequency(int arr[], int size, int n)
{
    int count = 0;
    for (int i=0; i<size; i++)
        if (arr[i] == n) ++count;
    return count;
}
void display(int a[],int n)
{
    cout<<"The random numbers are:"<<endl;
    for(int i=0; i<n; i++)
    {

        cout<<"a["<<i<<"]=";
        cout<<a[i];
        cout<<endl;
    }
}
void read(int arr[],int size)
{
    cout << "Enter " << size << " numbers:" << endl;
    for (int i = 0; i < size; ++i)
```

```
    {
        int a;
        a=rand()%999;
        arr[i]=a;
    }
}
```

Write a program to read and print a two-dimensional Matrix in C++(3x3 matrix)

```
#include <iostream>
using namespace std;
void read(int[][3]);
void print(int[][3]);
int main()
{
    int a[3][3];
    read(a);
    print(a);
}
void read(int a[][3])
{
    cout << "Enter 3 integers, 3 per row:\n";
         for (int i=0; i<3; i++)
    {
        for (int j=0; j<3; j++)
        {
            cout << "a[" << i << ", "<<j<<"]=";
                 cin >> a[i][j];
        }
    }
}
void print( int a[][3])
{
    cout << "The output is \n\n";
         for (int i=0; i<3; i++)
    {
        for (int j=0; j<3; j++)
        {
```

```
                cout<< a[i][j];
                cout<<'\t';
            }
            cout<<endl;
        }
}
```

Write a program to input numbers of 5 quiz of 3 student and find the average of marks obtained in each quiz and also average marks of each student.

```
#include <iostream>
using namespace std;
const int NUM_STUDENTS = 3;
const int NUM_QUIZZES = 5;
void read(int [][NUM_QUIZZES]);
void printQuizAverages(int [][NUM_QUIZZES]);
void printClassAverages(int [][NUM_QUIZZES]);
int main()
{
    int a[NUM_STUDENTS][NUM_QUIZZES];
    cout << "Enter " << NUM_QUIZZES << " scores for each student:\n";
    read(a);
    cout << "The quiz averages are:\n";
    printQuizAverages(a);
    cout << "The class averages are:\n";
    printClassAverages(a);
}
void read(int a[][NUM_QUIZZES])
{
    for (int s=0; s<NUM_STUDENTS; s++)
    {
        cout << "Student " << s << ":\n ";
        for (int q=0; q<NUM_QUIZZES; q++)
            {
                cout<<"Quiz "<<q<<" =";;
                cin >> a[s][q];
            }

    }
}
void printQuizAverages(int a[][NUM_QUIZZES])
{
    for (int s=0; s<NUM_STUDENTS; s++)
    {
        float sum = 0.0;
        for (int q=0; q<NUM_QUIZZES; q++)
            sum += a[s][q];
        cout << "\tStudent " << s << ": " << sum/NUM_QUIZZES << endl;
    }
}
void printClassAverages(int a[][NUM_QUIZZES])
{
    for (int q=0; q<NUM_QUIZZES; q++)
    {
        float sum = 0.0;
        for (int s=0; s<NUM_STUDENTS; s++)
            sum += a[s][q];
        cout << "\tQuiz " << q << ": " << sum/NUM_STUDENTS << endl;
    }
}
```

Write and test the function `void insert(float a[], int& n, float x)`
This function inserts the item x into the sorted array a of n elements and increments n. The new item is inserted at the location that maintains the sorted order of the array. This requires shifting elements forward to make room for the new x. (Note that this requires the array to have at least n+1 elements allocated.)

```
#include <iostream>
#include<cstdlib>
#include<ctime>
using namespace std;
void read(float[],int);
void insert(float [], int& , float );
void display(float[],int);
int main()
{
    srand(time(0));
    int size;int n;
    cout << "Enter the size of the array: ";
    cin >> size;
    int searchNum;
    cout << "Enter the number to search: ";
    cin >> searchNum;
    float arr[size];
    read(arr,size);
    display(arr,size);
    insert(arr,n,searchNum);
}
void display(float a[],int n)
{
    cout<<"The random numbers are:"<<endl;
    for(int i=0; i<n; i++)
    {

        cout<<"a["<<i<<"]=";
                        cout<<a[i];
        cout<<endl;
    }
}
void read(float arr[],int size)
{
    cout << "Enter " << size << " numbers:" << endl;
    for (int i = 0; i < size; ++i)
    {
        int a;
        a=rand()%10;
        arr[i]=a;
    }
}
void insert(float a[], int& n, float x)
{
    int j=n;
    while (j>0 && a[j-1]>x)
        a[j--] = a[j-1];
    a[j] = x;
    ++n;}
```

Write and test the following function: `void reverse(int a[], int n);`
The function reverses the first n elements of the array. For example, the call reverse(a,5) would transform the array {22,33,44,55,66,77,88,99} into {66,55,44,33,22,77,88,99}.

```
#include <iostream>
using namespace std;
void read(int[],int);
void reverse(int [], int );
void display(int[],int);
int main()
{

    int size;
    int n;
    cout << "Enter the size of the array: ";
    cin >> size;
    int arr[size];
    read(arr,size);
    display(arr,size);
    cout<<"Up to how many index to be reversed? ";
    cin>>n;
    reverse(arr,n);
    cout<<"After reverse the array is ";
    display(arr,size);
}
void display(int a[],int size)
{

    for(int i=0; i<size; i++)
    {

        cout<<"a["<<i<<"]=";
```

```
        cout<<a[i];
        cout<<endl;
    }
}
void read(int arr[],int size)
{
    cout << "Enter " << size << " numbers:" << endl;
    for (int i = 0; i < size; ++i)
    {
        cout<<"a["<<i<<"]=";
        cin>>arr[i];

    }
}

void reverse(int a[], int n)
{
    for (int i=0; i<n/2; i++)
       {
           int temp;
           temp=a[i];
           a[i]=a[n-1-i];
           a[n-1-i]=temp;
       }

}
```

Write and test the following **function:** `void add(float a[], int n, float b[]);` The **function** adds the first n elements of b to the corresponding first n elements of a. For example, **if** a is {2.2,3.3,4.4,5.5,6.6,7.7,8.8,9.9} and b is {6.0,5.0,4.0,3.0,2.0,1.0}, then the call add(a,5,b) would transform a into {8.2,8.3,8.4,8.5,8.6,7.7,8.8,9.9}.

```
#include <iostream>
using namespace std;
void read(float[],int);
void add(float [], int, float []);
void display(float[],int);
int main()
{

    int size,size1;
    int n;
    cout << "Enter the size of the array a : ";
    cin >> size;
     cout << "Enter the size of the array b : ";
    cin >> size1;
    float a[size];
    float b[size1];
    read(a,size);
    read(b,size1);
    cout<<"Up to how many index to be added? ";
    cin>>n;
    add(a,n,b);
    cout<<"After reverse the array is ";
    cout<<endl;
    display(a,size);
}
void display(float a[],int size)
{
```

```
    for(int i=0; i<size; i++)
    {

        cout<<"a["<<i<<"]=";
        cout<<a[i];
        cout<<endl;
    }
}
void read(float arr[],int size)
{
    cout << "Enter " << size << " numbers:" << endl;
    for (int i = 0; i < size; ++i)
    {
        cout<<"a["<<i<<"]=";
        cin>>arr[i];

    }
}


void add(float a[], int n, float b[])
{
    for (int i=0; i<n; i++)
        a[i] += b[i];
}
```

Write and test the **function** that "rotates" 90° clockwise a two-dimensional square array of ints. For example, it would transform the array
11 22 33
44 55 66
77 88 99
into the array
77 44 11
88 55 22
99 66 33

```
#include <iostream>
using namespace std;
void read(int[][3]);
void rotate(int [][3]);
void print(int[][3]);
int main()
{
    int a[3][3];
    read(a);
    cout<<"Before rotate\n";
     print(a);
    rotate(a);
    cout<<"After rotate\n";
    print(a);
}
void read(int a[][3])
{
    cout << "Enter 3 integers, 3 per row:\n";
    for (int i=0; i<3; i++)
    {
        for (int j=0; j<3; j++)
        {
            cout << "a[" << i << ", "<<j<<"]=";
            cin >> a[i][j];
        }
```

```
    }
}
void print( int a[][3])
{
    cout << "The output is \n\n";
    for (int i=0; i<3; i++)
    {
        for (int j=0; j<3; j++)
        {

            cout<< a[i][j];
            cout<<'\t';
        }
        cout<<endl;
    }
}
void rotate(int m[][3])
{
    int temp[3][3];
    for (int i=0; i<3; i++)
        for (int j=0; j<3; j++)
            temp[i][j] = m[3-j-1][i];
    for (int i=0; i<3; i++)
        for (int j=0; j<3; j++)
            m[i][j] = temp[i][j];
}
```