OOP LABORATORY 15

Name: **ANIRBAN HAZRA**
Section: **B-12**
Roll : **2005643**

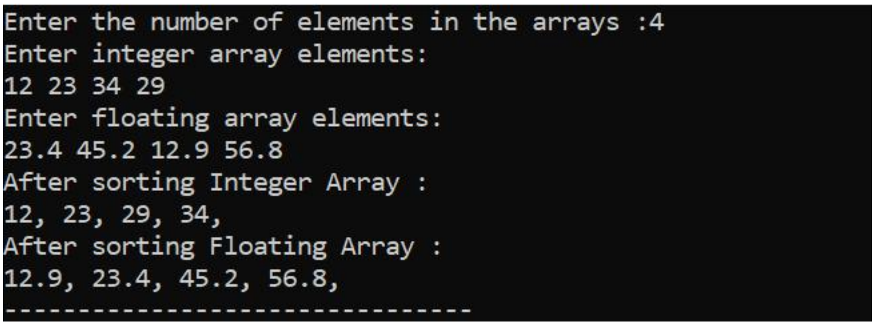
1. **WAP to find sort an integer array and a float array, using function template.**

PROGRAM CODE:

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

template <class T>
void sort(T arr[], int SIZE){
        for (int i = 0; i < SIZE; i++)
        {
                for (int j = i+1; j < SIZE; j++)
                {
                        if (arr[i] > arr[j])
                        {
                                T temp;
                                temp = arr[i];
                                arr[i] = arr[j];
                                arr[j] = temp;
                        }
                }
        }
}


int main(){
        int int_array[50], N;
        float float_array[50];

        cout<<"Enter the number of elements in the arrays :";
        cin>>N;

        cout<<"Enter integer array elements:"<<endl;
        for (int i = 0; i < N; i++)
        {
                cin>>int_array[i];
        }

        cout<<"Enter floating array elements:"<<endl;
        for (int i = 0; i < N; i++)
        {
                cin>>float_array[i];
        }

        sort(int_array,N);
```

```
    sort(float_array, N);

    cout<<"After sorting Integer Array :"<<endl;
    for (int i = 0; i < N; i++)
    {
        cout<<int_array[i]<<", ";
    }
    cout<<endl;

    cout<<"After sorting Floating Array :"<<endl;
    for (int i = 0; i < N; i++)
    {
        cout<<float_array[i]<<", ";
    }

    return 0;
}
```

OUTPUT:



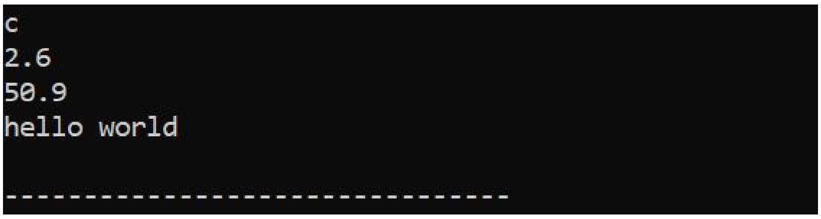
```
Enter the number of elements in the arrays :4
Enter integer array elements:
12 23 34 29
Enter floating array elements:
23.4 45.2 12.9 56.8
After sorting Integer Array :
12, 23, 29, 34,
After sorting Floating Array :
12.9, 23.4, 45.2, 56.8,
--------------------------------
```

2. WAP to display data of two different types using function template with multiple arguments.

PROGRAM CODE :

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

template<class T1, class T2> void display(T1 c1, T2 c2){
  cout<<c1<<endl;
  cout<<c2<<endl;
}

int main(){
  display('c',2.6);
  display(50.9,"hello world");
}
```

OUTPUT:



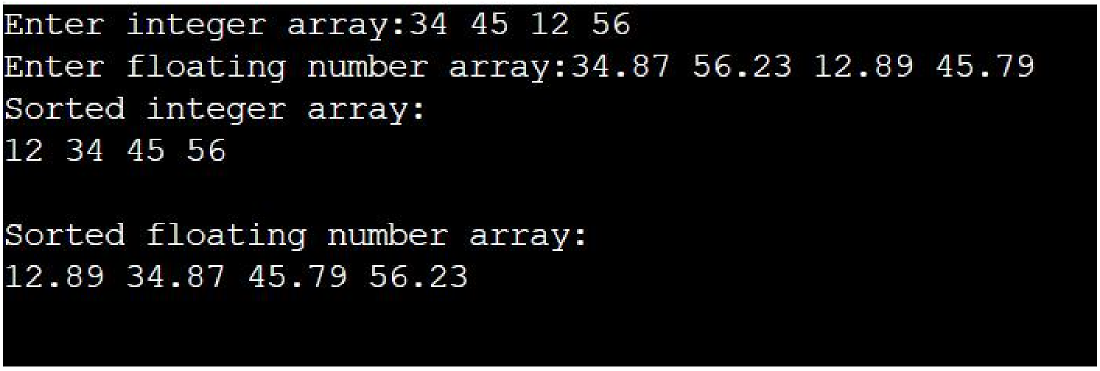
```
c
2.6
50.9
hello world

--------------------------------
```

3. Rewrite program i. using class template. Define the member functions outside the class

PROGRAM CODE:

```
#include<bits/stdc++.h>
using namespace std;
const int N = 4;

template <class Type>
class Array{
   private:
      Type arr[N];
   public:

      void read();
      void sortArr();
      void display();
};
template <class Type>
void Array<Type>::read(){
         for(int i = 0; i < N; i++){
            cin>>arr[i];
         }
      }
template <class Type>
void Array<Type>::sortArr(){
                              sort(arr,arr+N);
      }
template <class Type>
void Array<Type>::display(){
         int SIZE = sizeof(arr)/sizeof(Type);
         for(int i = 0; i < SIZE; i++){
            cout<<arr[i]<<" ";
         }
         cout<<endl;
      }

int main(){
   Array <int> i_obj;
   Array <float> f_obj;
   cout<<"Enter integer array:";
   i_obj.read();

   cout<<"Enter floating number array:";
   f_obj.read();
   i_obj.sortArr();
   f_obj.sortArr();
   cout<<"Sorted integer array:"<<endl;
   i_obj.display();
   cout<<endl<<"Sorted floating number array:"<<endl;
   f_obj.display();
   return 0;
}
```

OUTPUT:



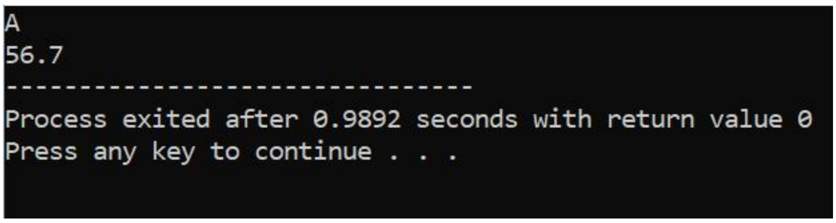
```
Enter integer array:34 45 12 56
Enter floating number array:34.87 56.23 12.89 45.79
Sorted integer array:
12 34 45 56

Sorted floating number array:
12.89 34.87 45.79 56.23
```

4. Rewrite program ii. using class template. Define the member functions outside the class

PROGRAM CODE:

```
#include<bits/stdc++.h>

using namespace std;

template <class T1 ,class T2>
class myclass
{
   public:
   T1 data1;
   T2 data2;
   myclass(T1 a,T2 b)
   {
     data1=a;
     data2=b;
   }
   void display();

};

template <class T1 ,class T2>
void myclass<T1,T2>::display()
   {
      cout<<this->data1<<endl<<this->data2;
   }


int main()
{
   myclass<char,float>obj('A',56.7);
   obj.display();
   return 0;
}
```

OUTPUT:



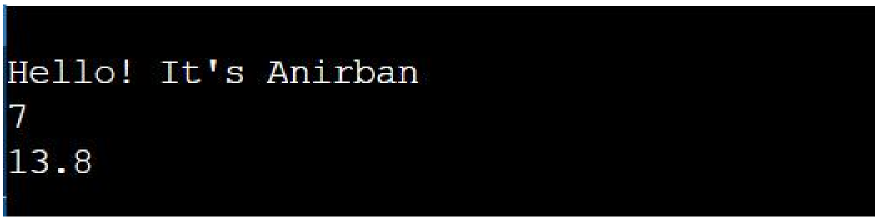
```
A
56.7
--------------------------------
Process exited after 0.9892 seconds with return value 0
Press any key to continue . . .
```

5. Write a function template to add two numbers. Overload the function template to add three numbers. The third template argument should have the default value as <int> . When the function is called with char type of arguments, the characters should be concatenated to form a string.

PROGRAM CODE:

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

template<class type>
void sum(type a, type b){
    type c ;
    c = a + b;
    cout<<"\n"<<c;
}

void sum(string a, string b){
    string c = a+b;
    cout<<"\n"<<c;
}

template<class type>
void sum(type a, type b, int n){
    type c ;
    c = a + b + n;
    cout<<"\n"<<c;
}

int main(){
    string x = "Hello! It's ";
    string y = "Anirban";
    sum(x,y);
    sum(3,4);
    sum(3.4,4.4,6);
    return 0;
}
```

OUTPUT:



```

Hello! It's Anirban
7
13.8
```