

Practice Lab Cycle Sheet

Object Oriented Programming

Submitted by: Hari Krishna Shah

VIT ID: 21BCS0167

Google Drive Link:

https://drive.google.com/drive/folders/1LeSJ6JuECLJR2tjcaWMz9N_xkzE8s4Ipp?usp=sharing

Ques 1 . Develop a function that returns through its reference parameters both the maximum and the minimum values stored in an array.

Answer:

```
#include <iostream>
#include <malloc.h>
using namespace std;

void calculate(int &maximum, int &minimum, int *array, int size);

int main(){
    int max, min, size, *array;
    cout<<"Enter the size of the array: ";
    cin>>size;

    array = (int *) (malloc(size * (sizeof(int))));
    cout<<"Enter the array elements : ";
    for(int i = 0; i<size; i++){
        cin>>array[i];
    }

    calculate(max, min, array, size);
    cout<<"The maximum value stored in the array is
"<<max<<"."<<endl;
    cout<<"The minimum value stored in the array is
```

```
"<<min<<"."<<endl;
```

```
return 0;
```

```
}
```

```
void calculate(int &maximum, int &minimum, int *array, int size){
```

```
    maximum = array[0];
```

```
    minimum = array[0];
```

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

```
        if(array[i]>maximum){
```

```
            maximum = array[i];
```

```
        }
```

```
    }
```

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

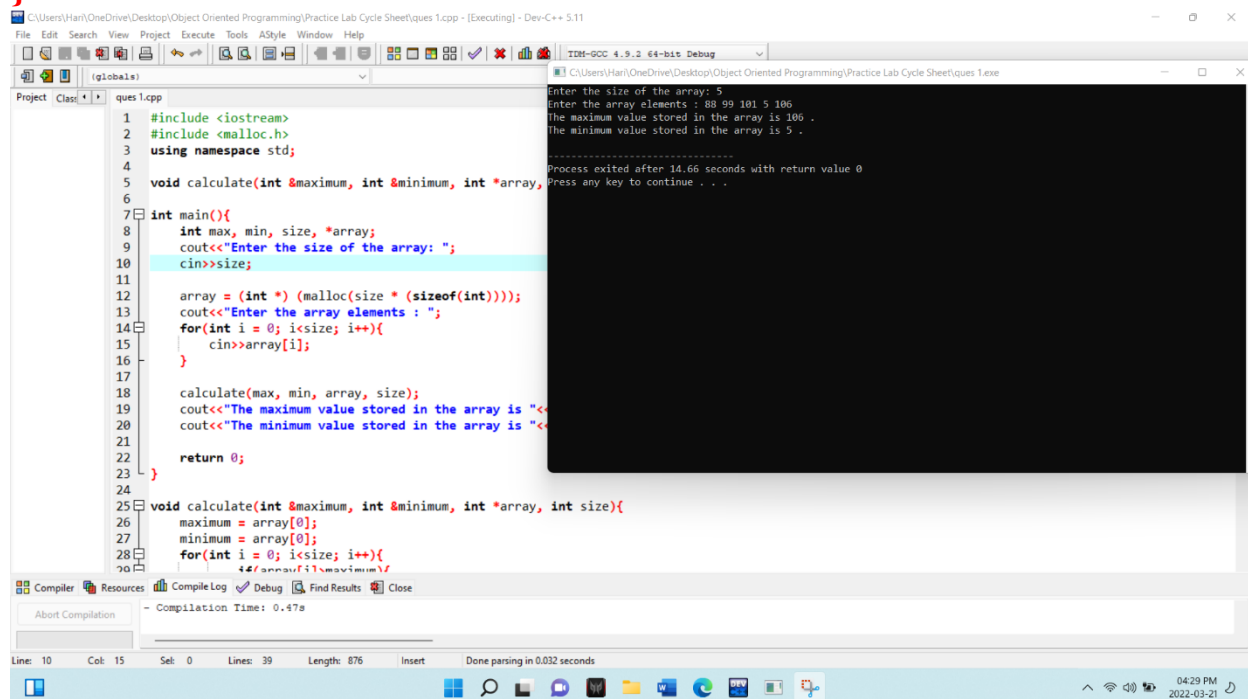
```
        if(array[i]<minimum){
```

```
            minimum = array[i];
```

```
        }
```

```
    }
```

```
}
```



```
1 #include <iostream>
2 #include <malloc.h>
3 using namespace std;
4
5 void calculate(int &maximum, int &minimum, int *array,
6
7 int main(){
8     int max, min, size, *array;
9     cout<<"Enter the size of the array: ";
10    cin>>size;
11
12    array = (int *) (malloc(size * (sizeof(int))));
13    cout<<"Enter the array elements : ";
14    for(int i = 0; i<size; i++){
15        cin>>array[i];
16    }
17
18    calculate(max, min, array, size);
19    cout<<"The maximum value stored in the array is "<<
20    cout<<"The minimum value stored in the array is "<<
21
22    return 0;
23 }
24
25 void calculate(int &maximum, int &minimum, int *array, int size){
26     maximum = array[0];
27     minimum = array[0];
28     for(int i = 0; i<size; i++){
29         if(array[i]>maximum){
30             maximum = array[i];
31         }
32     }
33     for(int i = 0; i<size; i++){
34         if(array[i]<minimum){
35             minimum = array[i];
36         }
37     }
38 }
```

Enter the size of the array: 5
Enter the array elements : 88 99 101 5 106
The maximum value stored in the array is 106 .
The minimum value stored in the array is 5 .
.....
Process exited after 14.66 seconds with return value 0
Press any key to continue . . .

Compiler Resources Compile Log Debug Find Results Close
- Compilation Time: 0.47s

Line: 10 Col: 15 Sel: 0 Lines: 39 Length: 876 Insert Done parsing in 0.032 seconds

04:29 PM
2022-03-21

Ques 2. Implement and test the following function:

bool isPrime(int num); // returns true iff num is a prime //

examples: isPrime(17) returns true; // isPrime(100) returns false.

Answer:

```
#include <iostream>
```

```
#include <stdbool.h>
```

```
using namespace std;
```

```
//Coded by Hari Krishna Shah
```

```
bool isprime(int num);
```

```
int main(){
```

```
    int num;
```

```
    char option;
```

```
    bool prime;
```

```
    do{
```

```
        cout<<"Enter a number: ";
```

```
        cin>>num;
```

```
        prime = isprime(num);
```

```
        if(prime){
```

```
            cout<<"The number "<<num<<" is a prime  
number."<<endl;
```

```
        }
```

```
        else{
```

```
            cout<<"The number "<<num<<" is not a prime  
number."<<endl;
```

```
        }
```

```
        cout<<"\nEnter q to quit the program or any other key  
to run the program again: ";
```

```
        cin>>option;
```

```
        cout<<endl;
```

```
    }
```

```
    while(option != 'q');
```

```
    cout<<"Thank you for using the program."<<endl;
```

```
    return 0;
```

```
}
```

```
bool isprime(int num){
```

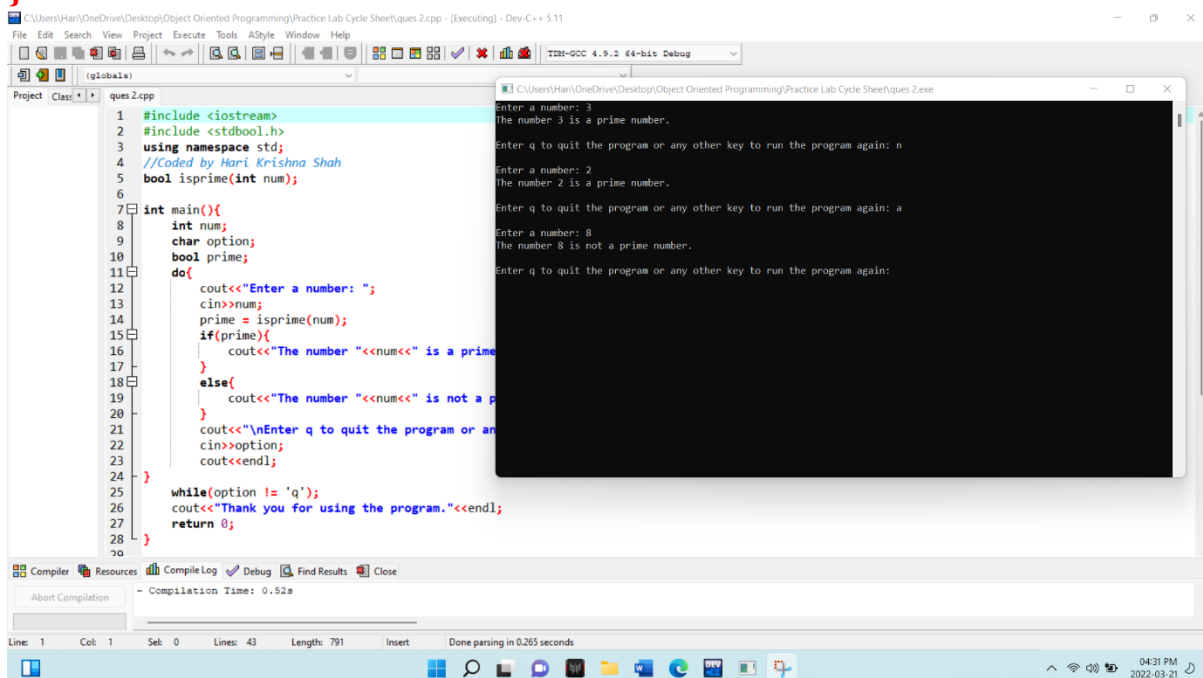
```

int count = 0;
for(int i = 1; i<=num; i++){
    if(num %i == 0){
        count ++;
    }
}

if(count == 2){
    return true;
}

return false;
}

```



Ques 3. Write a function called “maxOfThree” to find the largest of three integer values passed to it and return the largest value. Overload the same function accepting double values and strings.

Answer:

```
#include <iostream>
```

```
#include <string>
```

```

#include<cstring>
using namespace std;
int compare(int a, int b, int c);
double compare(double A, double B, double C);
string &compare(string a1, string b2, string c3);

int main(){
    int option = 0;
    while(option != -1){
        cout<<"Enter an option from the menu below.\n \
        Enter 1 to compare three integers\n \
        Enter 2 to compare three doubles\n \
        Enter 3 to compare three strings\n \
        Enter -1 to quite the program\n";
        cout<<"Enter your option here: ";
        cin>>option;
        switch(option){
            case 1:{
                int a, b, c, largest_int;
                cout<<"Enter three integers: ";
                cin>>a>>b>>c;
                largest_int = compare(a, b, c);
                cout<<"The largest integer is
"<<largest_int<<"."<<endl;
                break;

```

```

    }
    case 2:{
        double A, B, C, largest_double;
        cout<<"Enter three doubles: ";
        cin>>A>>B>>C;
        largest_double = compare(A, B, C);
        cout<<"The largest double is
"<<largest_double<<"."<<endl;
        break;
    }
    case 3:{
        string a1, b2, c3, largest;
        cout<<"Enter three strings separated by
whitespace: ";

        cin>>a1>>b2>>c3;
        largest = compare(a1, b2, c3);
        cout<<"The largest string is "<<largest<<"
."<<endl;

        break;
    }
    default:
        cout<<"Enter a valid option and try
again."<<endl;
    }
    cout<<"\n"<<endl;
}

```

```

}
int compare(int a, int b, int c){
    int large;
    large = (a>b)?(a>c?a:c): (b>c ? b: c);
    return large;
}
double compare(double A, double B, double C){
    int large;
    large = (A>B)?(A>C?A:C): (B>C ? B: C);
    return large;
}
string &compare(string a1, string b2, string c3){
    if(a1.compare(b2)>0){
        if(a1.compare(c3)>0){
            return a1;
        }
        else{
            return c3;
        }
    }
    else{
        if(b2.compare(c3)>0){
            return b2;
        }
        else{

```

```

        return c3;
    }
}
}

```

```

1 #include <iostream>
2 #include <string>
3 #include <cstring>
4 using namespace std;
5 int compare(int a, int b, int c);
6 double compare(double A, double B, double C);
7 string &compare(string a1, string b2, string c3);
8
9 int main(){
10     int option = 0;
11     while(option != -1){
12         cout<<"Enter an option from the menu below.\n";
13         cout<<"Enter 1 to compare three integers\n \n";
14         cout<<"Enter 2 to compare three doubles\n \n";
15         cout<<"Enter 3 to compare three strings\n \n";
16         cout<<"Enter -1 to quite the program\n";
17         cout<<"Enter your option here: ";
18         cin>>option;
19         switch(option){
20             case 1:{
21                 int a, b, c, largest_int;
22                 cout<<"Enter three integers: ";
23                 cin>>a>>b>>c;
24                 largest_int = compare(a, b, c);
25                 cout<<"The Largest integer is "<<largest_int<<endl;
26             }
27             case 2:{
28                 double A, B, C, largest_double;
29                 cout<<"Enter three doubles: ";
30                 cin>>A>>B>>C;
31                 largest_double = compare(A, B, C);
32                 cout<<"The Largest double is "<<largest_double<<endl;
33             }
34             case 3:{
35                 string a1, b2, c3, largest_string;
36                 cout<<"Enter three strings separated by whitespace: a b c ";
37                 cin>>a1>>b2>>c3;
38                 largest_string = &compare(a1, b2, c3);
39                 cout<<"The Largest string is "<<largest_string<<endl;
40             }
41             default:
42                 cout<<"Invalid option\n";
43         }
44     }
45 }

```

Output Window:

```

Select C:\Users\Har\OneDrive\Desktop\Object Oriented Programming\Practice Lab Cycle Sheet\ques 3.exe
Enter three integers: 45 95 20
The largest integer is 95 .

Enter an option from the menu below.
Enter 1 to compare three integers
Enter 2 to compare three doubles
Enter 3 to compare three strings
Enter -1 to quite the program
Enter your option here: 2
Enter three doubles: 88 99 102
The largest double is 102 .

Enter an option from the menu below.
Enter 1 to compare three integers
Enter 2 to compare three doubles
Enter 3 to compare three strings
Enter -1 to quite the program
Enter your option here: 3
Enter three strings separated by whitespace: a b c
The largest string is c .

Enter an option from the menu below.
Enter 1 to compare three integers
Enter 2 to compare three doubles
Enter 3 to compare three strings
Enter -1 to quite the program
Enter your option here:

```

Ques 4. Create a C++ class called Book containing bookTitle, author(s), publication, yearOfPublication and price as data members. Write functions for the following operations and test them by creating an array of Book objects. Also include member functions in the same class for input and output of data members and perform the following operations. i) Display the books name sorted in ascending order. ii) Display the books detail authored by “XXX” (“XXX” must be accepted from user) iii) Display the details of books whose price is greater than 2000.

Answer:

```
#include <iostream>
```

```
#include <cstring>
```

```
#include <string>
```



```
#include <malloc.h>

using namespace std;

class books{
    private:
        char title[50];
        char author[50];
        char publication[50];
        int year_pub;
        float price;
    public:
        void get_details();
        string return_title();
        void display();
        string return_author();
        int return_price();

};

void books::get_details(){
    cout<<"Enter the title of the book: ";
    cin>>title;
    cout<<"Enter the author of the book: ";
    cin>>author;
    cout<<"Enter the publisher of the book: ";
    cin>>publication;
```

```
        cout<<"Enter the year of publication of the book: ";
        cin>>year_pub;

        cout<<"Enter the price of the book: ";
        cin>>price;
    }
```

```
void books::display(){
    cout<<"Book Title: "<<title<<endl;
    cout<<"Author: "<<author<<endl;
    cout<<"Publisher: "<<publication<<endl;
    cout<<"Year of publication: "<<year_pub<<endl;
    cout<<"Price: "<<price<<endl<<endl;
}
```

```
string books::return_title(){
    return title;
}
```

```
string books::return_author(){
    return author;
}
```

```
int books::return_price(){
    return price;
}
```

```
int main(){
```

```

class books *b;
b = (class books *) (malloc(sizeof(class books)));
static int book_count = 0;
int temp1_count = 0, temp2_count = 0;
int option;
do{
    cout<<"\t\tWelcome to the main menu"<<endl;
    cout<<"Enter an option from the menu below\n \
Enter 1 to add books to the database\n \
Enter 2 to display all the books\n \
Enter 3 to sort the book in ascending order\n \
Enter 4 to search the book by author name\n \
Enter 5 to display books whose price is below certain
price"<<endl;
    cout<<"Enter your option here: ";
    cin>>option;
    cout<<endl;
    switch(option){
        case 1:{
            cout<<"Enter the number of books you want to
add: ";
            cin>>temp1_count;
            book_count += temp1_count;
            cout<<endl;
            for(; temp2_count<book_count;
temp2_count++){

```

```

        cout<<"Enter the details for book number
"<<temp2_count+1<<" : "<<endl;

        b = (class books *) (realloc(b,
book_count*sizeof(class books)));

        b[temp2_count].get_details();
        cout<<endl;

    }
    break;
}

case 2:{
    if(book_count == 0){
        cout<<"Add some books first."<<endl;
        break;
    }
    for(int i = 0; i<book_count; i++){
        cout<<"Details for book number
"<<i+1<<" is given below."<<endl;
        b[i].display();
        cout<<endl;
    }
    break;
}

case 3:{
    if(book_count == 0){
        cout<<"There's isn't any book in the
database. Add some books to the database first."<<endl;

```

```

        break;
    }
    class books temp;
    for(int i = 0; i<book_count; i++){
        for(int j = i+1; j<book_count; j++){

            if((b[i].return_title().compare(b[j].return_title()))>0){

                temp = b[i];
                b[i] = b[j];
                b[j] = temp;

            }

        }

        cout<<"The books have been sorted in
ascending order according to book title successfully."<<endl;
        break;
    }

    case 4:{
        if(book_count == 0){
            cout<<"There's isn't any book in the
database. Add some books to the database first."<<endl;
            break;
        }
        char search[50];

```

```

        cout<<"Enter the author name to search in the
database: ";

        cin>>search;
        int flag1 = 0;
        for(int i = 0; i<book_count; i++){

            if(b[i].return_author().compare(search)==0){

                cout<<"Following book is authored
by "<<search<<" .\nBook number: "<<flag1+1<<"."<<endl;

                b[i].display();

                cout<<endl;

                flag1 += 1;

            }

        }

        if(flag1 == 0){

            cout<<"There isn't any book authored by
"<<search<<" in the database.\nTry again with another author
name."<<endl;

        }

        else{

            cout<<"Total "<<flag1<<" numbers of
books were found in the database."<<endl;

        }

        break;

    }

```

case 5:

```

        {
            if(book_count == 0){
                cout<<"There's isn't any book in the
database. Add some books to the database first."<<endl;
                break;
            }
            int price;
            cout<<"Enter the price to find books below that
price: ";

            cin>>price;
            int flag = 0;
            for(int i = 0; i<book_count; i++){
                if(b[i].return_price()<price){
                    cout<<"Match Found. Book
number: "<<flag+1<<endl;

                    b[i].display();
                    flag += 1;
                }
            }
            if(flag == 0){
                cout<<"There isn't any book below
Rs."<<price<<" in the database."<<endl;
            }
            break;
        }
    }

```

```

    }

    cout<<endl;

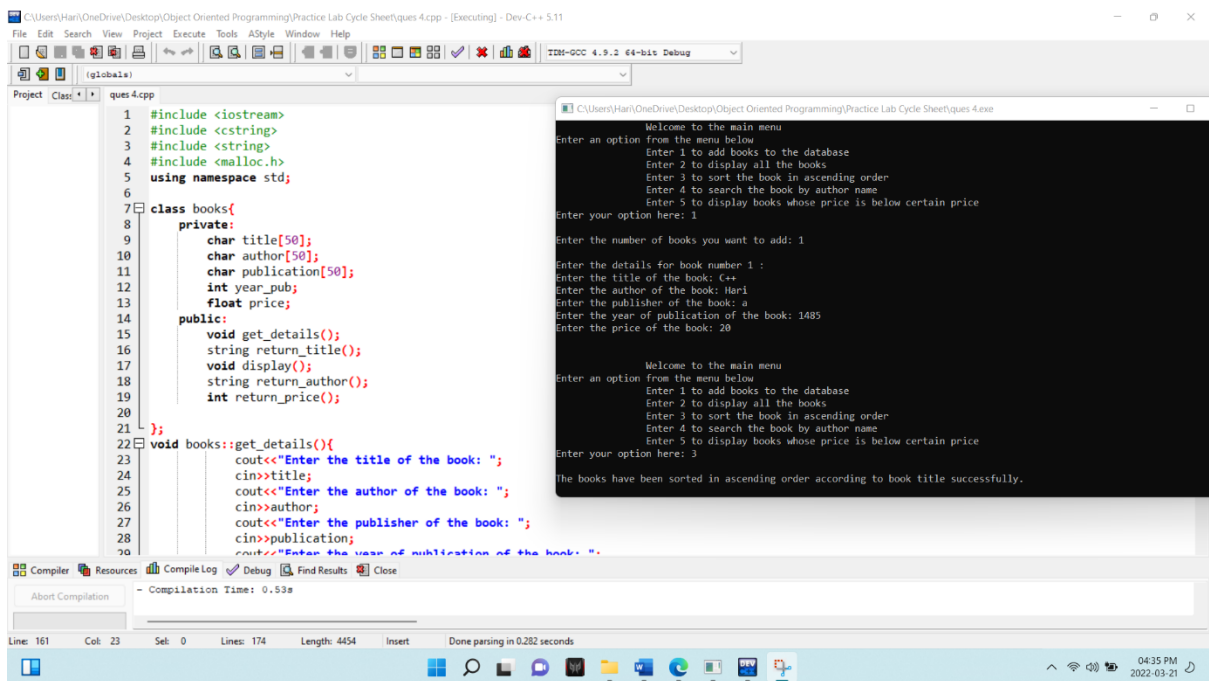
}

while(option != -1);

return 0;

}

```



Ques 5.

```
/*
```

Write a C++ program to illustrate the usage of function call operator as lvalue in an assignment statement. (Note: Return By Reference)


```
*/
```

```
#include <iostream>
```

```
using namespace std;
```

```
int & compare(int *a, int *b){
```

```
    if(*a>*b){
```

```
        return *a;
```

```
    }
```

```
    else{
```

```
        return *b;
```

```
    }
```

```
}
```

```
int main(){
```

```
    int x, y;
```

```
    cout<<"Enter the two integer values: ";
```

```
    cin>>x>>y;
```

```
    cout<<"The maximum value is "<<compare(&x, &y)<<"  
.\n"<<endl;
```

```
    cout<<"Now, the value of variable with the maximum value is  
replaced by 100 using return by reference below.\n"<<endl;
```

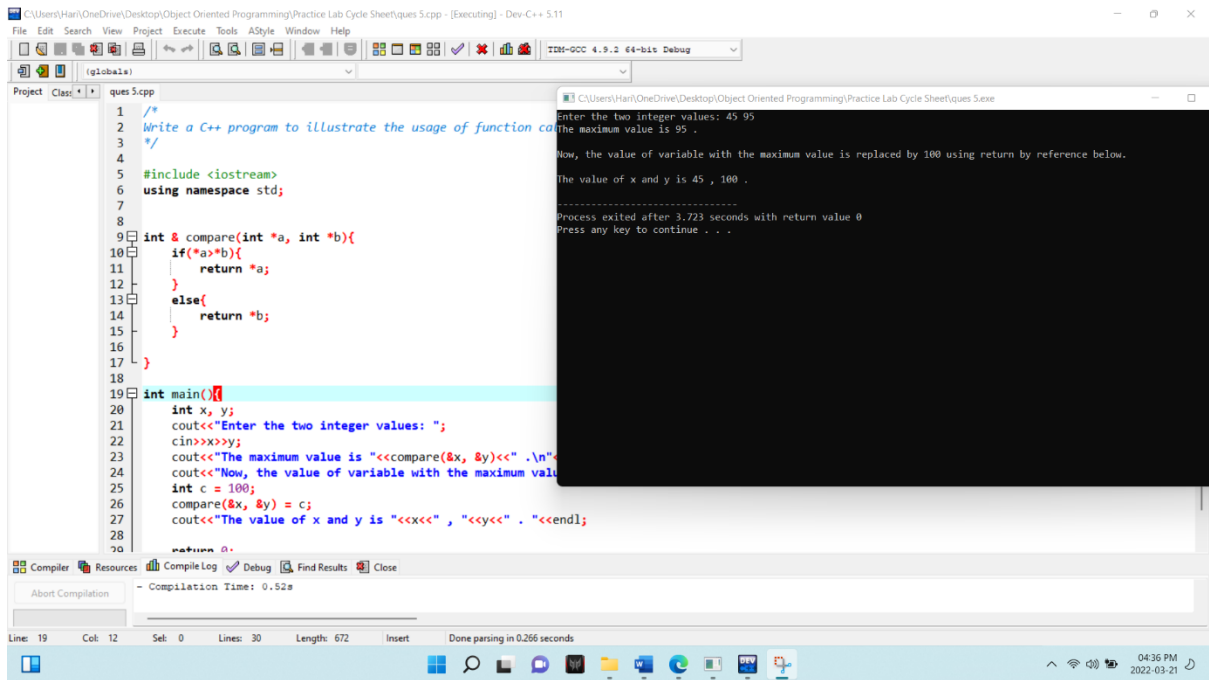
```
    int c = 100;
```

```
compare(&x, &y) = c;
```

```
cout<<"The value of x and y is "<<x<<" , "<<y<<" . "<<endl;
```

```
return 0;
```

```
}
```



```
1  /*
2  Write a C++ program to illustrate the usage of function call
3  */
4
5  #include <iostream>
6  using namespace std;
7
8
9  int & compare(int *a, int *b){
10     if(*a>*b){
11         return *a;
12     }
13     else{
14         return *b;
15     }
16 }
17
18
19 int main(){
20     int x, y;
21     cout<<"Enter the two integer values: ";
22     cin>>x>>y;
23     cout<<"The maximum value is "<<compare(&x, &y)<<" .\n";
24     cout<<"Now, the value of variable with the maximum value is replaced by 100 using return by reference below."<<endl;
25     int c = 100;
26     compare(&x, &y) = c;
27     cout<<"The value of x and y is "<<x<<" , "<<y<<" . "<<endl;
28 }
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

Ques 6

/*Write a C++ function called “zstrcmp” to compare one string with another string or compare first ‘n’ characters of one string with another string.

Use default arguments and Do not overload the function. “zstrcmp” function must return an integer which has the same interpretation as that of “strcmp” in C. */

```
#include <iostream>
```

```
using namespace std;
```

//Without using inbuilt function.

```
int zstrcmp(char *string1, char *string2, int c_size);
```

```
int main(){
```

```
    char str1[100], str2[100];
```

```
    int c_size, compare_value;
```

```
    cout<<"Enter the first string: ";
```

```
    cin>>str1;
```

```
    cout<<"Enter the second string: ";
```

```
    cin>>str2;
```

```
    cout<<"Enter the number of characters you want to compare: ";
```

```
    cin>>c_size;
```

```
    compare_value = zstrcmp(str1, str2, c_size);
```

```
    if(compare_value == 0){
```

```
        cout<<"The strings are equal."<<endl;
```

```
    }
```

```
    if(compare_value == 1){
```

```
        cout<<"The string1 i.e "<<str1<<" is greater than the  
string2 i.e "<<str2<<" . "<<endl;
```

```

    }
    if(compare_value == -1){
        cout<<"The string2 i.e "<<str2<<" is greater than the
string1 i.e "<<str1<<"."<<endl;
    }

    return 0;
}

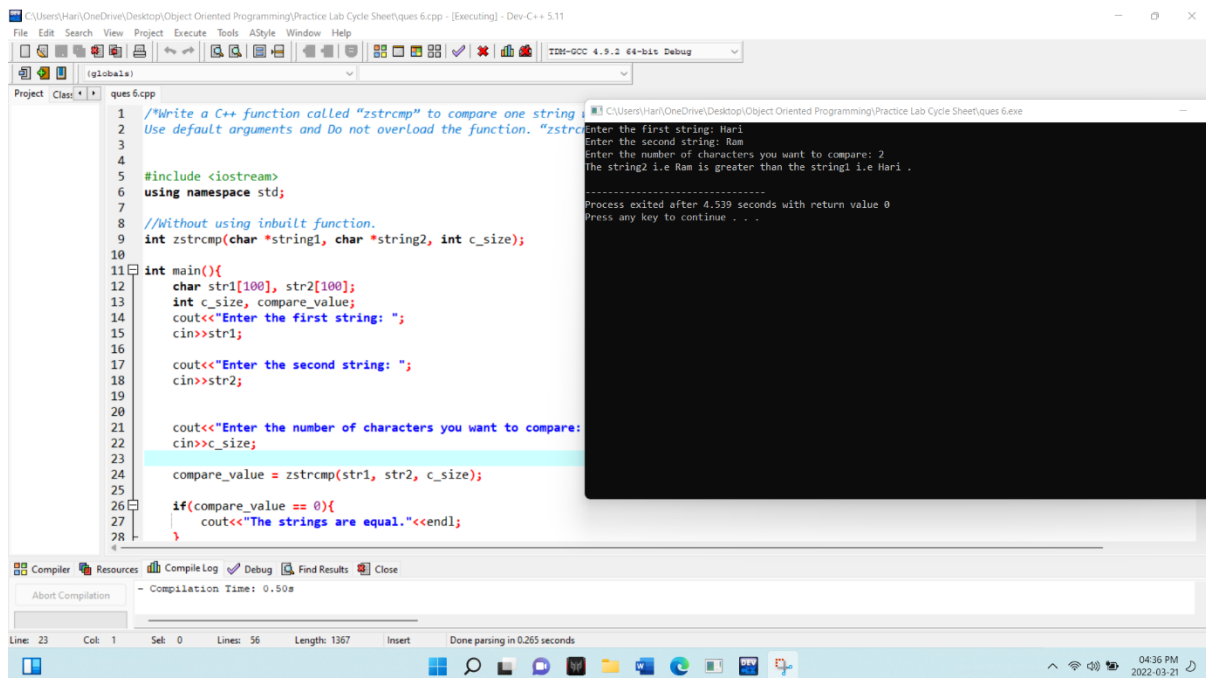
```

```

int zstrcmp(char *string1, char *string2, int c_size){
    for(int i = 0; i<c_size; i++){
        if(*string1 != '\0' && *string2 != '\0'){
            if(*string1>*string2){
                return 1;
            }
            else{
                return -1;
            }
        }
        *string1 += 1;
        *string2 += 1;
    }
    return 0;
}

```

}



Ques 7.

/*4. Write a C++ function called "zstrcpy" to copy one string to another

or copy first 'n' characters of one string into another. Use default arguments

and Do not overload the function.

*/

```
#include <iostream>
```

```
using namespace std;
```

```
void ztrcpy(char st1[50], char str2[50]);
```

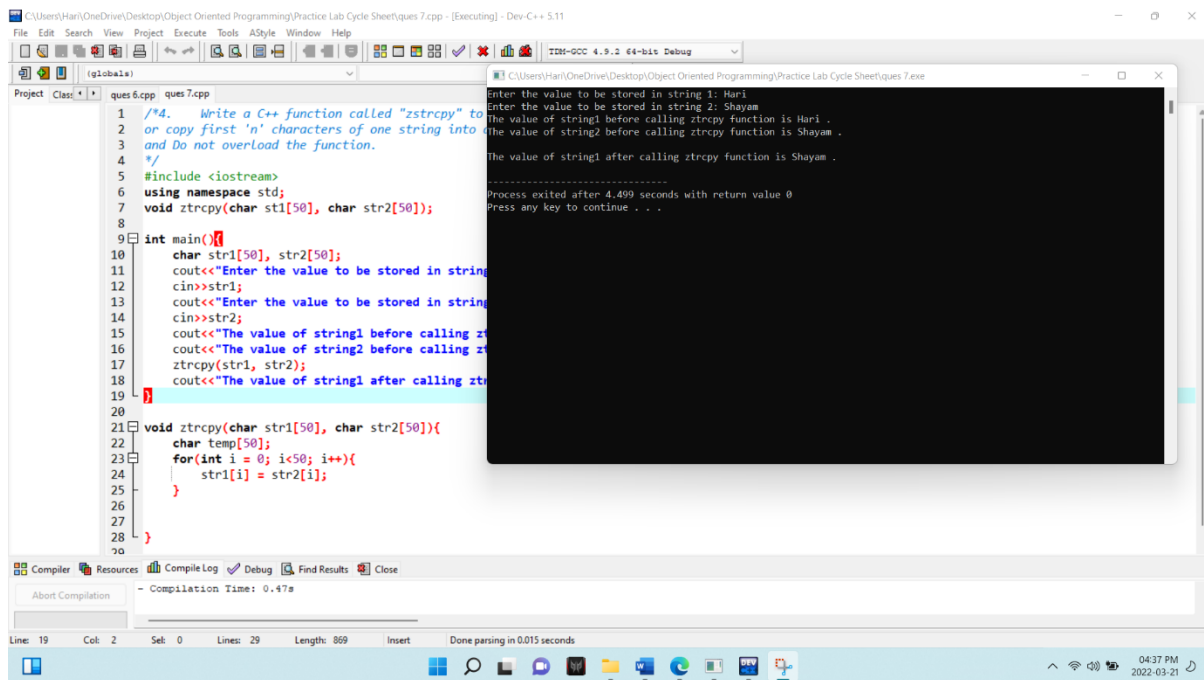
```
int main(){
```

```
    char str1[50], str2[50];
```

```
    cout<<"Enter the value to be stored in string 1: ";
```

```
    cin>>str1;
    cout<<"Enter the value to be stored in string 2: ";
    cin>>str2;
    cout<<"The value of string1 before calling ztrcpy function is
"<<str1<<"."<<endl;
    cout<<"The value of string2 before calling ztrcpy function is
"<<str2<<"."<<endl<<endl;
    ztrcpy(str1, str2);
    cout<<"The value of string1 after calling ztrcpy function is
"<<str1<<"."<<endl;
}
```

```
void ztrcpy(char str1[50], char str2[50]){
    char temp[50];
    for(int i = 0; i<50; i++){
        str1[i] = str2[i];
    }
}
```



Ques 8.

```
#include <iostream>
```

```
using namespace std;
```

```
void search(int x, int array[], int size);
```

```
void search(char x, char string[], int size);
```

```
void search(float x, float f_array[], int size);
```

```
int main(){
```

```
    int size1 = 0, size2 = 0, size3 = 0;
```

```
    int integer;
```

```
    char character;
```

```
    float floats;
```

```
    int array[size1];
```

```
    char string[size2];
```

```
    float f_array[size3];
```

```
cout<<"Enter the size of int array: ";
cin>>size1;
cout<<"Enter the elements of the int array: ";
for(int i = 0; i<size1; i++){
    cin>>array[i];
}
cout<<"Enter the integer to search: ";
cin>>integer;
search(integer, array, size1);
cout<<"Enter the size of char array: ";
cin>>size2;
cout<<"Enter the elements of the character array: ";
for(int i = 0; i<size2; i++){
    cin>>string[i];
}
cout<<"Enter the character to search: ";
cin>>character;
search(character, string, size2);
cout<<"Enter the size of float array: ";
cin>>size3;
cout<<"Enter the elements of the float array: ";
for(int i = 0; i<size2; i++){
    cin>>f_array[i];
}
```



```

        cout<<"Enter the float to search: ";
        cin>>floats;
        search(floats, f_array, size3);
    }

```

```

void search(int x, int array[], int size){
    int flag = 0;
    for(int i = 0; i<size; i++){
        if(array[i] == x){
            cout<<"The searched integer is available at the index
"<<i<<" .\n\n"<<endl;
            flag += 1;
        }
    }
    if(flag == 0){
        cout<<"The searched element was not found in the
array.\n\n"<<endl;
    }
}

```

```

void search(char x, char string[], int size){
    int flag = 0;
    for(int i = 0; i<size; i++){
        if(string[i] == x){
            cout<<"The searched character is available at the
index "<<i<<" .\n\n"<<endl;
            flag += 1;
        }
    }
}

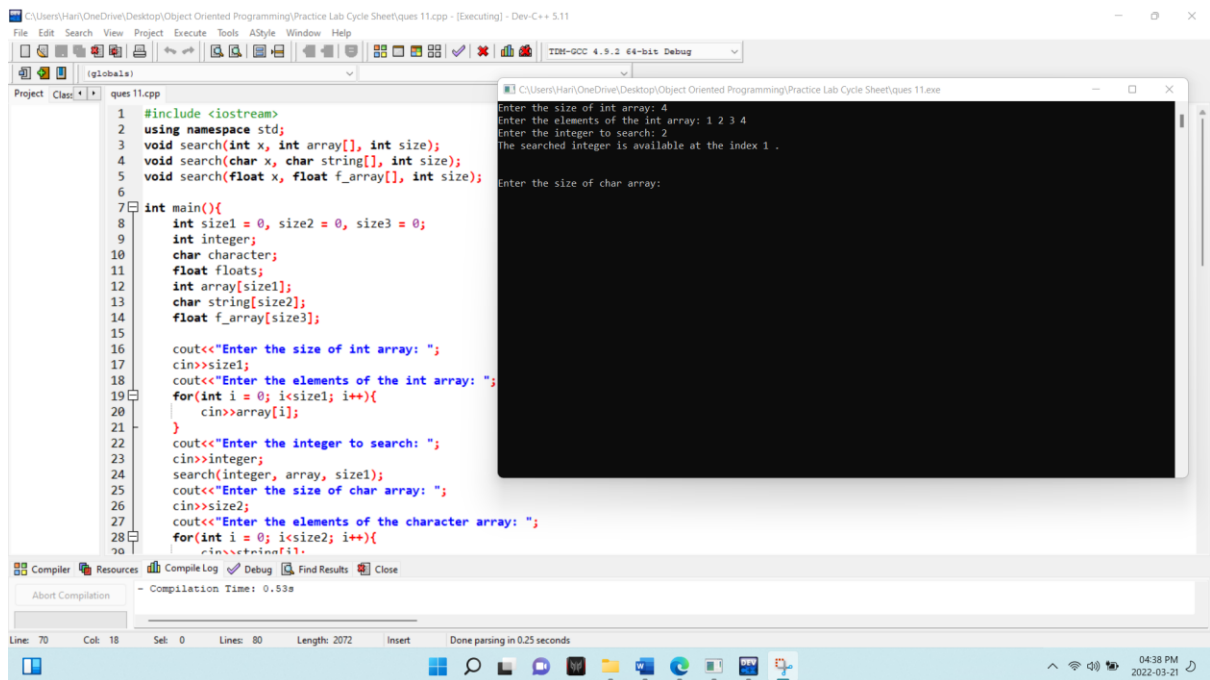
```

```

        }
    }
    if(flag == 0){
        cout<<"The searched element was not found in the
array.\n\n"<<endl;
    }
}

void search(float x, float f_array[], int size){
    int flag = 0;
    for(int i = 0; i<size; i++){
        if(f_array[i] == x){
            cout<<"The searched float is available at the index
"<<i<<" .\n\n"<<endl;
            flag += 1;
        }
    }
    if(flag == 0){
        cout<<"The searched element was not found in the
array.\n\n"<<endl;
    }
}

```



Ques 9.

```
#include <iostream>
```

```
using namespace std;
```

```
void sort(int array[], int size);
```

```
void sort(char string[], int size);
```

```
void sort(float f_array[], int size);
```

```
int main(){
```

```
    int size1 = 0, size2 = 0, size3 = 0;
```

```
    int integer;
```

```
    char character;
```

```
    float floats;
```

```
    int array[size1];
```

```
    char string[size2];
```

```
float f_array[size3];

cout<<"Enter the size of int array: ";
cin>>size1;
cout<<"Enter the elements of the int array: ";
for(int i = 0; i<size1; i++){
    cin>>array[i];
}
cout<<"The elements of the unsorted array are: ";
for(int i = 0; i<size1; i++){
    cout<<array[i]<<" ";
}
cout<<endl;
sort(array, size1);
cout<<"The elements of the sorted array are: ";
for(int i = 0; i<size1; i++){
    cout<<array[i]<<" ";
}
cout<<endl<<endl;

cout<<"Enter the size of char array: ";
cin>>size2;
cout<<"Enter the elements of the character array: ";
for(int i = 0; i<size2; i++){
    cin>>string[i];
```

```
}  
cout<<"The elements of the unsorted array are: ";  
for(int i = 0; i<size2; i++){  
    cout<<string[i]<<" ";  
}  
cout<<endl;  
sort(string, size2);  
cout<<"The elements of the sorted array are: ";  
for(int i = 0; i<size2; i++){  
    cout<<string[i]<<" ";  
}  
cout<<endl<<endl;
```

```
cout<<"Enter the size of float array: ";  
cin>>size3;  
cout<<"Enter the elements of the float array: ";  
for(int i = 0; i<size2; i++){  
    cin>>f_array[i];  
}  
cout<<"The elements of the unsorted array: ";  
for(int i = 0; i<size2; i++){  
    cout<<f_array[i];  
}  
cout<<endl;  
sort(f_array, size3);
```

```

        cout<<"The elements of the sorted array: ";
        for(int i = 0; i<size2; i++){
            cout<<f_array[i]<<endl;
        }
        cout<<endl;
        return 0;
    }

void sort(int array[], int size){
    int temp;
    for(int i = 0; i<size; i++){
        for(int j = i+1; j<size; j++){
            if(array[i]>array[j]){
                temp = array[i];
                array[i] = array[j];
                array[j] = temp;
            }
        }
    }
}

void sort(char string[], int size){
    char temp;
    for(int i = 0; i<size; i++){
        for(int j = i+1; j<size; j++){
            if(string[i]>string[j]){
                temp = string[i];

```

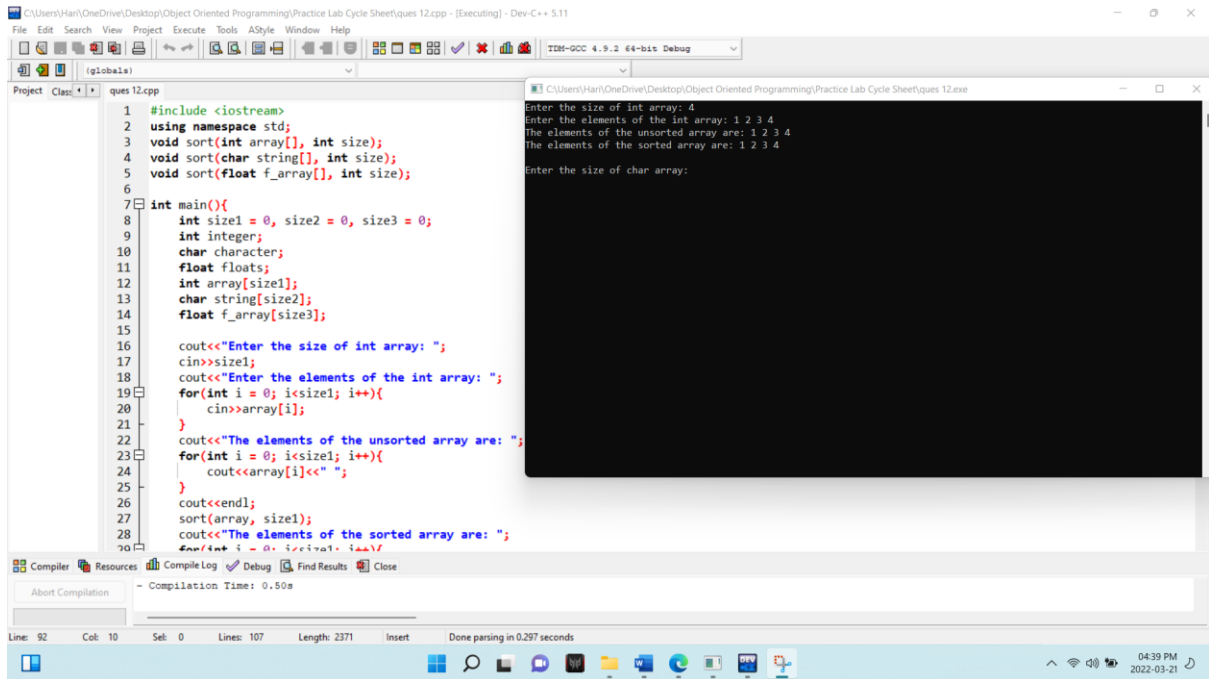
```

        string[i] = string[j];
        string[j] = temp;
    }
}
}

}

void sort(float f_array[], int size){
    float temp;
    for(int i = 0; i<size; i++){
        for(int j = i+1; j<size; j++){
            if(f_array[i]>f_array[j]){
                temp = f_array[i];
                f_array[i] = f_array[j];
                f_array[j] = temp;
            }
        }
    }
}

```



Ques 10.

```
/*
```

Write a recursive function to multiply two integers.

```
*/
```

```
#include <iostream>
```

```
using namespace std;
```

```
int multiplication(int x, int y);
```

```
int main(){
```

```
    int a, b, multi;
```

```
    cout<<"Enter the value for two integers: ";
```

```
    cin>>a>>b;
```

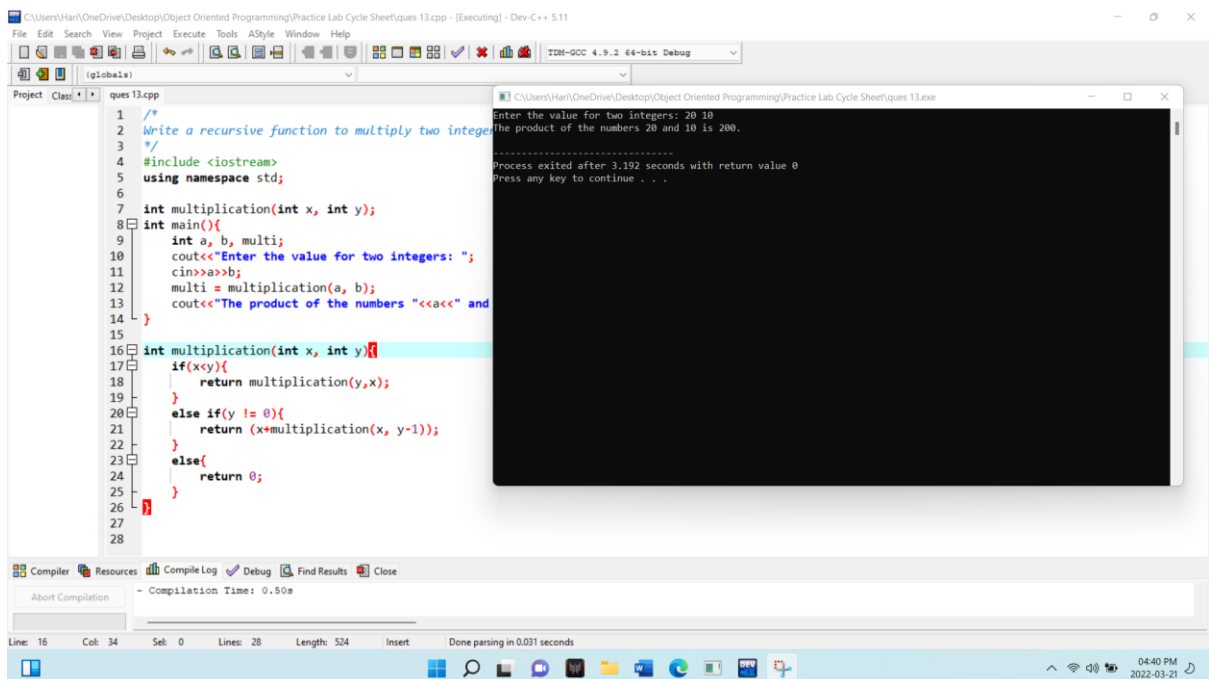
```
    multi = multiplication(a, b);
```

```
    cout<<"The product of the numbers "<<a<<" and "<<b<<" is
"<<multi<<". "<<endl;
```



```
}
```

```
int multiplication(int x, int y){  
    if(x<y){  
        return multiplication(y,x);  
    }  
    else if(y != 0){  
        return (x+multiplication(x, y-1));  
    }  
    else{  
        return 0;  
    }  
}
```



Ques 11

```
/*
```

Write a recursive function to find the sum of first 'N' Natural numbers.

```
*/
```

```
#include <iostream>
```

```
using namespace std;
```

```
int sum(int num);
```

```
int main(){
```

```
    int num, total;
```

```
    cout<<"Enter a number: ";
```

```
    cin>>num;
```

```
    total = sum(num);
```

```
    cout<<"The sum of the first natural numbers upto number  
"<<num<<" is "<<total<<"."<<endl;
```

```
}
```

```
int sum(int num){
```

```
    if(num == 0){
```

```
        return 0;
```

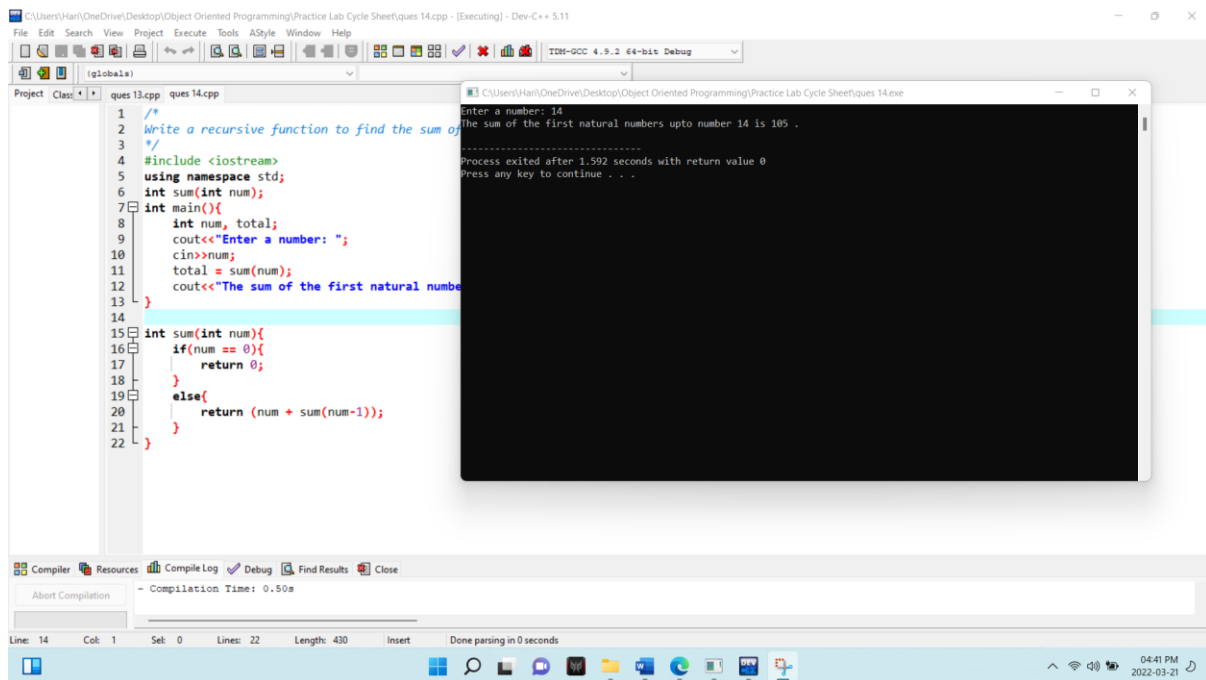
```
    }
```

```
    else{
```

```
        return (num + sum(num-1));
```

```
    }
```

```
}
```



Ques 12.

```
/*
```

Write a recursive function to calculate the Fibonacci value.

```
*/
```

```
#include <iostream>
```

```
using namespace std;
```

```
int fibonacci_value(int index);
```

```
int main(){
```

```
    int count, index = 0;
```

```
    cout<<"Enter the number of fibonacci values you want: ";
```

```
    cin>>count;
```

```
    cout<<"The fibonacci series is: ";
```

```
    for(int i = 1; i<=count; i++){  
        cout<<fibonacci_value(index)<<" ";  
        index++;  
    }  
    return 0;  
}
```

```
int fibonacci_value(int index){  
    if(index == 0){  
        return 0;  
    }  
    else if(index == 1){  
        return 1;  
    }  
    else{  
        return (fibonacci_value(index-1)+fibonacci_value(index-  
2));  
    }  
}
```

