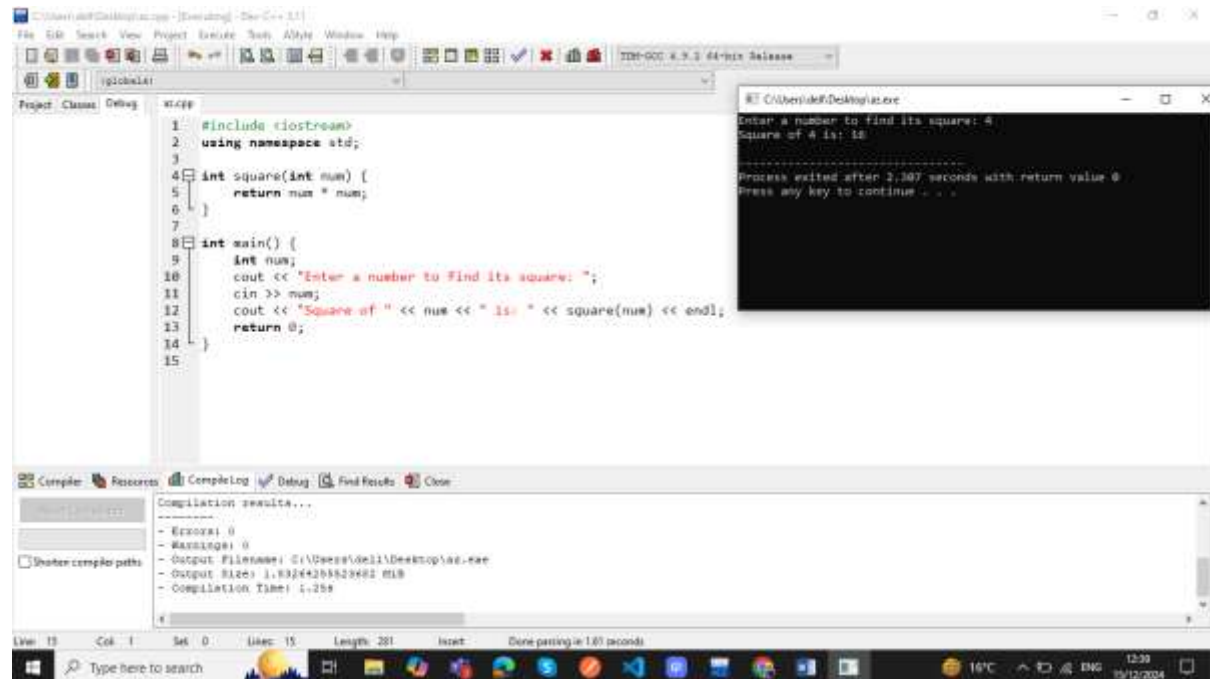


NAME:ZOHA

SAPID:65341

Qusetion#1



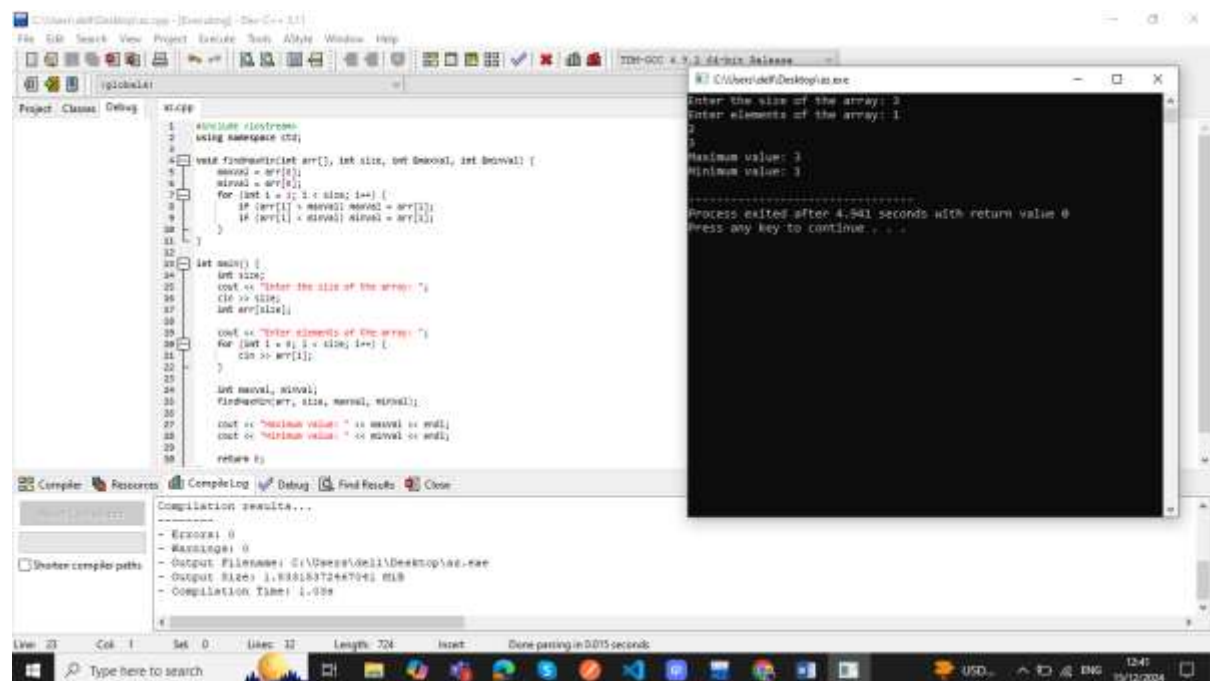
The screenshot shows a C++ IDE with a project named "qt.cpp". The code defines a function `square` that takes an integer `num` and returns `num * num`. The `main` function prompts the user to enter a number, reads the input, and prints the square. The output window shows the user entered 4, and the program output is "Square of 4 is: 16". The compilation results show 0 errors and 0 warnings.

```
1 #include <iostream>
2 using namespace std;
3
4 int square(int num) {
5     return num * num;
6 }
7
8 int main() {
9     int num;
10    cout << "Enter a number to find its square: ";
11    cin >> num;
12    cout << "Square of " << num << " is: " << square(num) << endl;
13    return 0;
14 }
15
```

Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\deli\Desktop\ad.exe
- Output Size: 1.8344265828682 MB
- Compilation Time: 1.25s

QUESTION#2



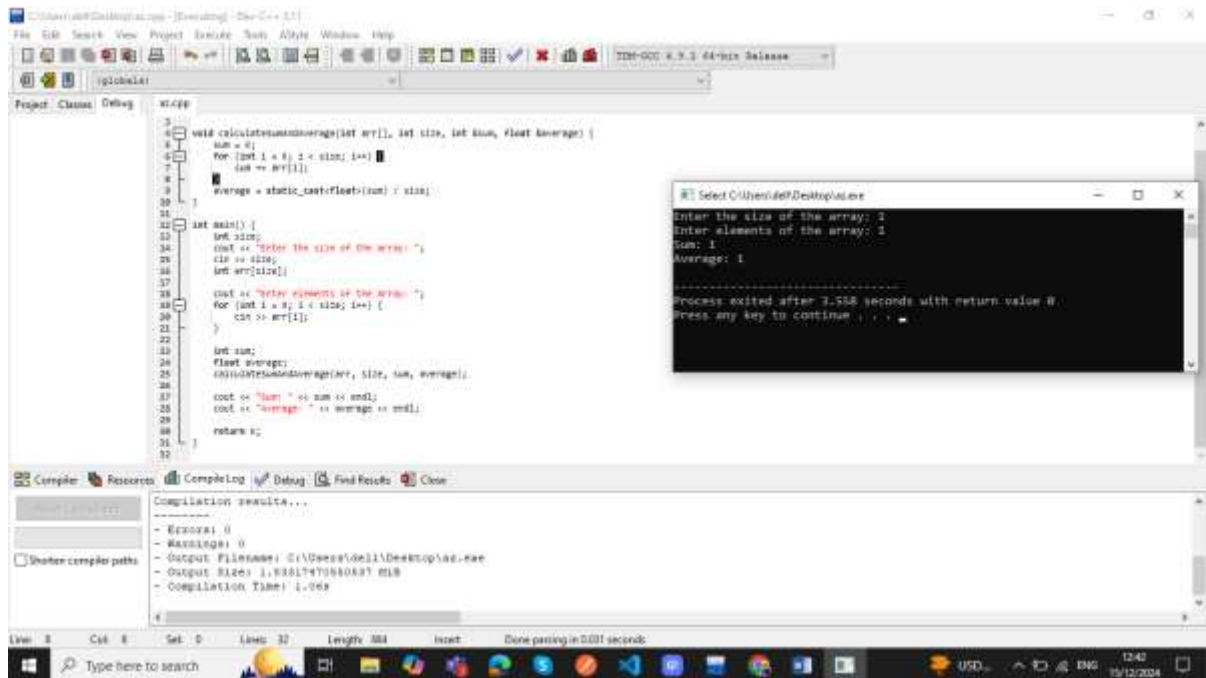
The screenshot shows a C++ IDE with a project named "qt.cpp". The code defines a function `findmaxmin` that takes an array `arr`, its size `size`, and pointers to store the maximum and minimum values. The `main` function prompts the user to enter the size of the array and the elements, then calls `findmaxmin` and prints the maximum and minimum values. The output window shows the user entered 5 for the size and 1 2 3 4 5 for the elements, and the program output is "Maximum value: 5" and "Minimum value: 1". The compilation results show 0 errors and 0 warnings.

```
1 #include <iostream>
2 using namespace std;
3
4 void findmaxmin(int arr[], int size, int *maxval, int *minval) {
5     *maxval = arr[0];
6     *minval = arr[0];
7     for (int i = 1; i < size; i++) {
8         if (arr[i] > *maxval) *maxval = arr[i];
9         if (arr[i] < *minval) *minval = arr[i];
10    }
11 }
12
13 int main() {
14    int size;
15    cout << "Enter the size of the array: ";
16    cin >> size;
17    int arr[size];
18    cout << "Enter elements of the array: ";
19    for (int i = 0; i < size; i++) {
20        cin >> arr[i];
21    }
22    int maxval, minval;
23    findmaxmin(arr, size, &maxval, &minval);
24    cout << "Maximum value: " << maxval << endl;
25    cout << "Minimum value: " << minval << endl;
26    return 0;
27 }
28
```

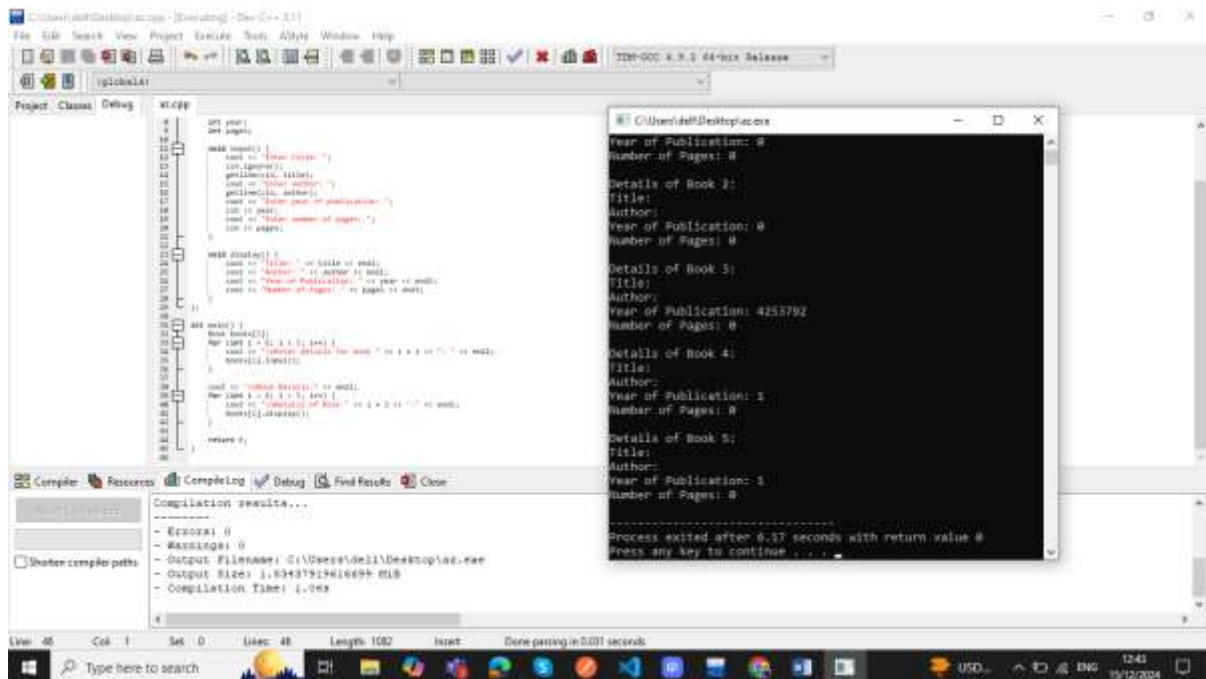
Compilation results...

- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\deli\Desktop\ad.exe
- Output Size: 1.8348372447941 MB
- Compilation Time: 1.99s

### QUESTION#3



### QUESTION#4



## QUESTION#5

The screenshot shows a C++ IDE with a file named `login.cpp`. The code defines a `User` struct with a `name` string and a `main` function that prompts for a username. If the username is not empty, it prints a welcome message; otherwise, it prints a login failure message.

```

1 #include <iostream>
2 #include <string>
3 using namespace std;
4
5 struct User {
6     string name;
7 };
8
9 int main() {
10     User user;
11     cout << "Enter your username: ";
12     cin >> user.name;
13
14     if (user.name != "") {
15         cout << "Welcome, " << user.name << "!" << endl;
16     } else {
17         cout << "Login failed." << endl;
18     }
19
20     return 0;
21 }

```

The output window shows the program's execution: it prompts for a username, receives "EGH", and prints "Welcome, EGH!". The process exits after 3.898 seconds with a return value of 0.

Compilation results show 0 errors and 0 warnings. The output file is `C:\Users\delil\Desktop\login.exe`, with a size of 1,834,879,437,288 MB and a compilation time of 1.96s.

## QUESTION#6

The screenshot shows a C++ IDE with a file named `student.cpp`. The code defines a `Student` struct with fields for name, ID, address, department, marks, and maximum marks. It includes functions for adding, deleting, and displaying student records.

```

1 #include <iostream>
2 #include <string>
3 #include <vector>
4 using namespace std;
5
6 struct Student {
7     string name;
8     int ID;
9     string address;
10    string department;
11    int marks;
12    int maximum_marks;
13 };
14
15 void addStudent(vector<Student> &students) {
16     Student s;
17     cout << "Enter name: ";
18     getline(cin, s.name);
19     cout << "Enter ID: ";
20     cin >> s.ID;
21     cout << "Enter address: ";
22     getline(cin, s.address);
23     cout << "Enter department: ";
24     getline(cin, s.department);
25     cout << "Enter marks: ";
26     cin >> s.marks;
27     cout << "Enter maximum marks: ";
28     cin >> s.maximum_marks;
29     students.push_back(s);
30 }
31
32 void deleteStudent(vector<Student> &students) {
33     int i;
34     for (i = 0; i < students.size(); i++) {
35         cout << "Enter details for student: " << i << " " << endl;
36         cout << "Name: " << students[i].name << endl;
37         cout << "ID: " << students[i].ID << endl;
38         cout << "Address: " << students[i].address << endl;
39         cout << "Department: " << students[i].department << endl;
40         cout << "Marks: " << students[i].marks << endl;
41         cout << "Maximum Marks: " << students[i].maximum_marks << endl;
42     }
43     if (i == 0) {
44         cout << "No student found." << endl;
45     }
46 }
47
48 void displayStudent(vector<Student> &students) {
49     cout << "Displaying students:" << endl;
50     for (int i = 0; i < students.size(); i++) {
51         cout << "Enter details for student: " << i << " " << endl;
52         cout << "Name: " << students[i].name << endl;
53         cout << "ID: " << students[i].ID << endl;
54         cout << "Address: " << students[i].address << endl;
55         cout << "Department: " << students[i].department << endl;
56         cout << "Marks: " << students[i].marks << endl;
57         cout << "Maximum Marks: " << students[i].maximum_marks << endl;
58     }
59 }
60
61 int main() {
62     vector<Student> students;
63     addStudent(students);
64     deleteStudent(students);
65     displayStudent(students);
66 }

```

The output window shows the program's execution: it prompts for student details, displays the details, and then exits after 48.4 seconds with a return value of 0.

Compilation results show 0 errors and 0 warnings. The output file is `C:\Users\delil\Desktop\student.exe`, with a size of 1,834,879,437,288 MB and a compilation time of 1.96s.