Name : Muhid Qaiser

Roll Number : 22i-0472

Section : AI-B

Report

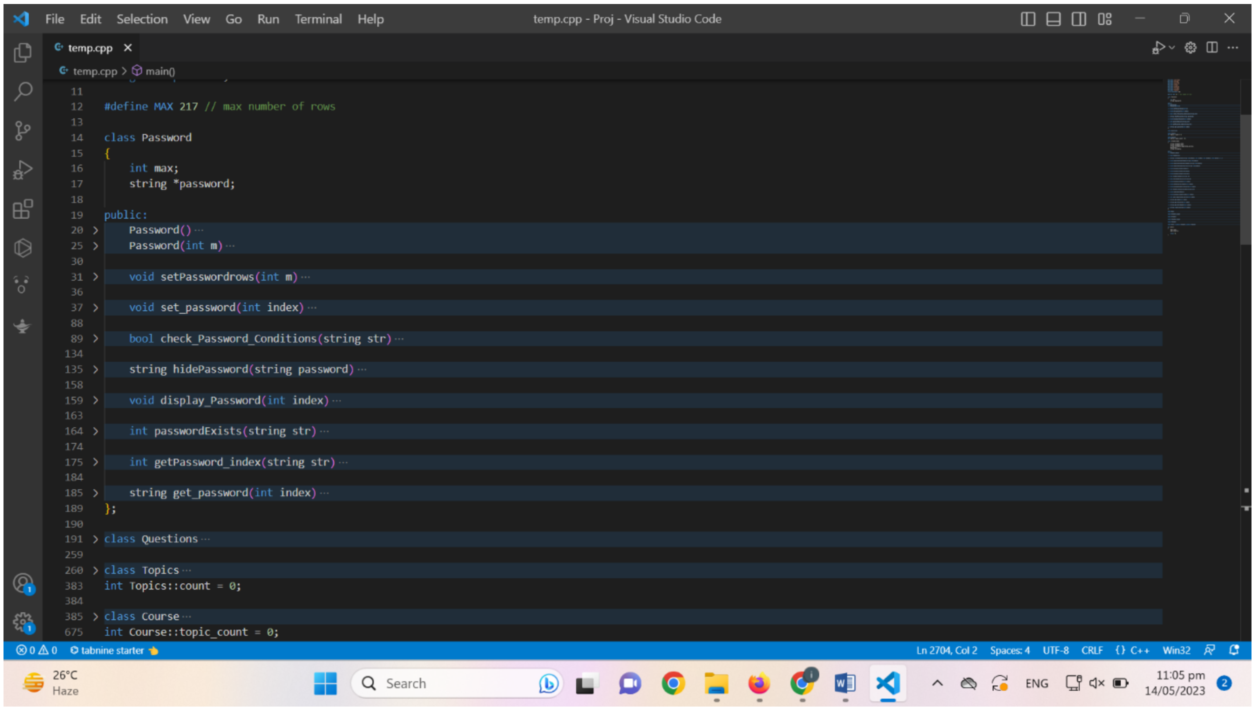
Application of OOP in the System.

This system utilizes OOP concepts in a great deal to function. Concepts ranging from Association, Aggregation to Inheritance etc. there are a number of classes in the system that utilizes such concepts.

Lets start with,

Class Password :

This class is used to set and manage passwords of students. It also checks for password validty and returns them.



```cpp

#define MAX 217 // max number of rows

class Password
{
    int max;
    string *password;

public:
    Password() ···
    Password(int m) ···

    void setPasswordrows(int m) ···

    void set_password(int index) ···

    bool check_Password_Conditions(string str) ···

    string hidePassword(string password) ···

    void display_Password(int index) ···

    int passwordExists(string str) ···

    int getPassword_index(string str) ···

    string get_password(int index) ···
};

class Questions ···

class Topics ···
int Topics::count = 0;

class Course ···
int Course::topic_count = 0;
```

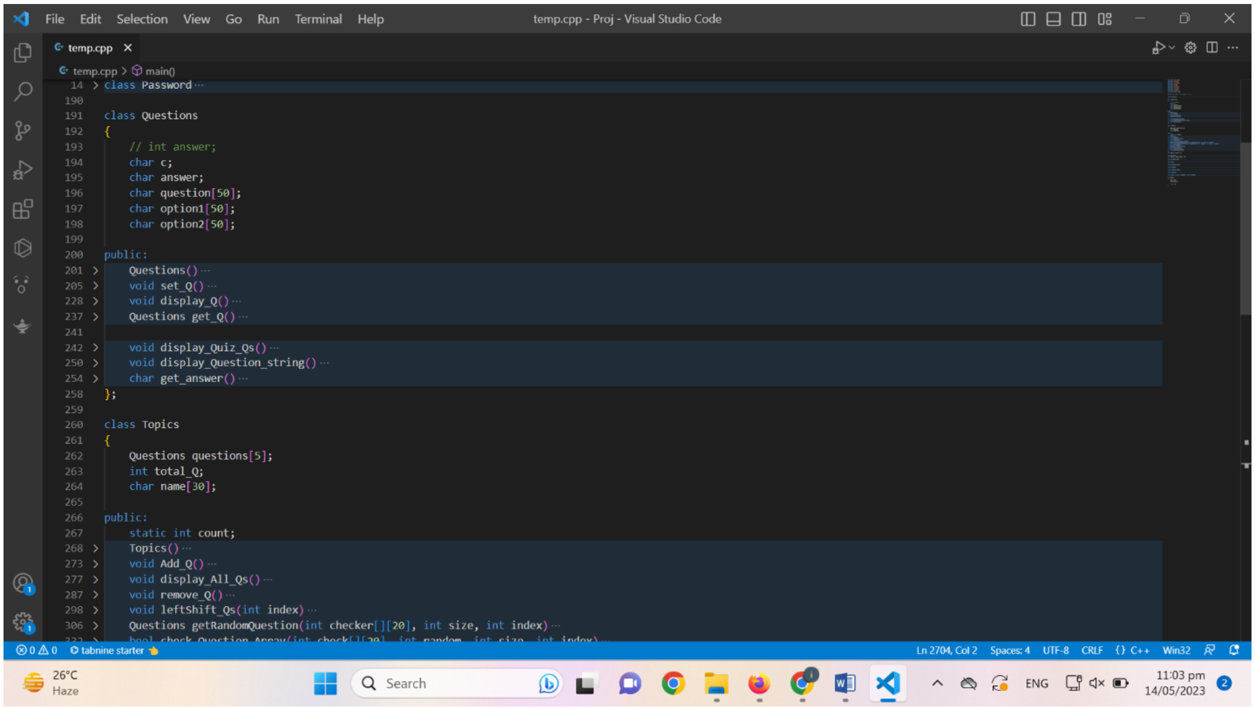
Class Questions :

It consist of char arrays for storing and displaying questions and their options and answers. This class also returns a Questions object.

Class Topics :

This class has a Composition relationship with class Questions. Topics has many Questions.

This is because Topics class consists of an array of Questions without which It cannot exist. It uses these Question arrays to add Questions in each indexes and keep track of which Question is in which index. It also deletes, displays and returns Questions. This class contains static variables to keep track of returning Questions

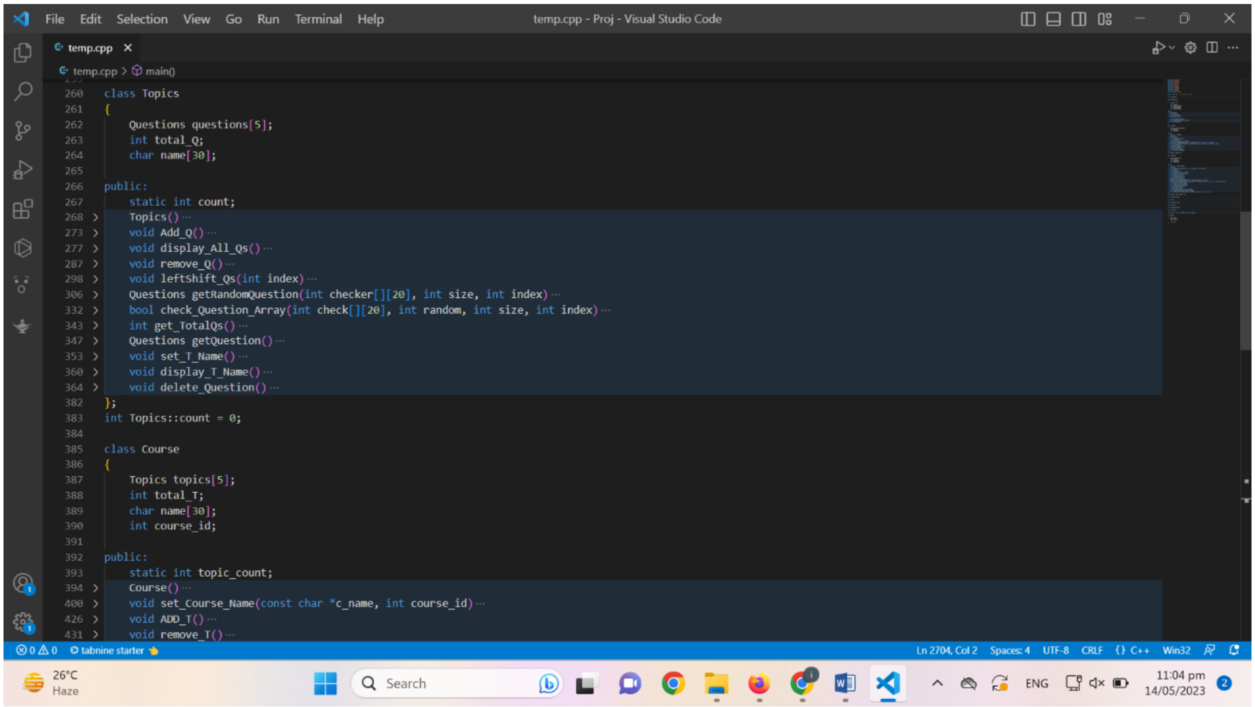


```
class Password···

class Questions
{
    // int answer;
    char c;
    char answer;
    char question[50];
    char option1[50];
    char option2[50];

public:
    Questions()···
    void set_Q()···
    void display_Q()···
    Questions get_Q()···

    void display_Quiz_Qs()···
    void display_Question_string()···
    char get_answer()···
};

class Topics
{
    Questions questions[5];
    int total_Q;
    char name[30];

public:
    static int count;
    Topics()···
    void Add_Q()···
    void display_All_Qs()···
    void remove_Q()···
    void leftShift_Qs(int index)···
    Questions getRandomQuestion(int checker[][20], int size, int index)···
    bool check_Question_Array(int check[][20], int random, int size, int index)···
```

Class Course :

This class has a Composition relationship with class Topics. Course has many Topics.

This is because Course class consists of an array of Topics without which It cannot exist. It uses these Topics arrays to add Topics and Questions in each indexes and keep track of which Topic is in which index. It also deletes, displays and returns Questions. . This class contains static variables to keep track of returning Questions



```
class Topics
{
    Questions questions[5];
    int total_Q;
    char name[30];

public:
    static int count;
    Topics() ···
    void Add_Q() ···
    void display_All_Qs() ···
    void remove_Q() ···
    void leftShift_Qs(int index) ···
    Questions getRandomQuestion(int checker[][20], int size, int index) ···
    bool check_Question_Array(int check[][20], int random, int size, int index) ···
    int get_TotalQs() ···
    Questions getQuestion() ···
    void set_T_Name() ···
    void display_T_Name() ···
    void delete_Question() ···
};
int Topics::count = 0;

class Course
{
    Topics topics[5];
    int total_T;
    char name[30];
    int course_id;

public:
    static int topic_count;
    Course() ···
    void set_Course_Name(const char *c_name, int course_id) ···
    void ADD_T() ···
    void remove_T() ···
```

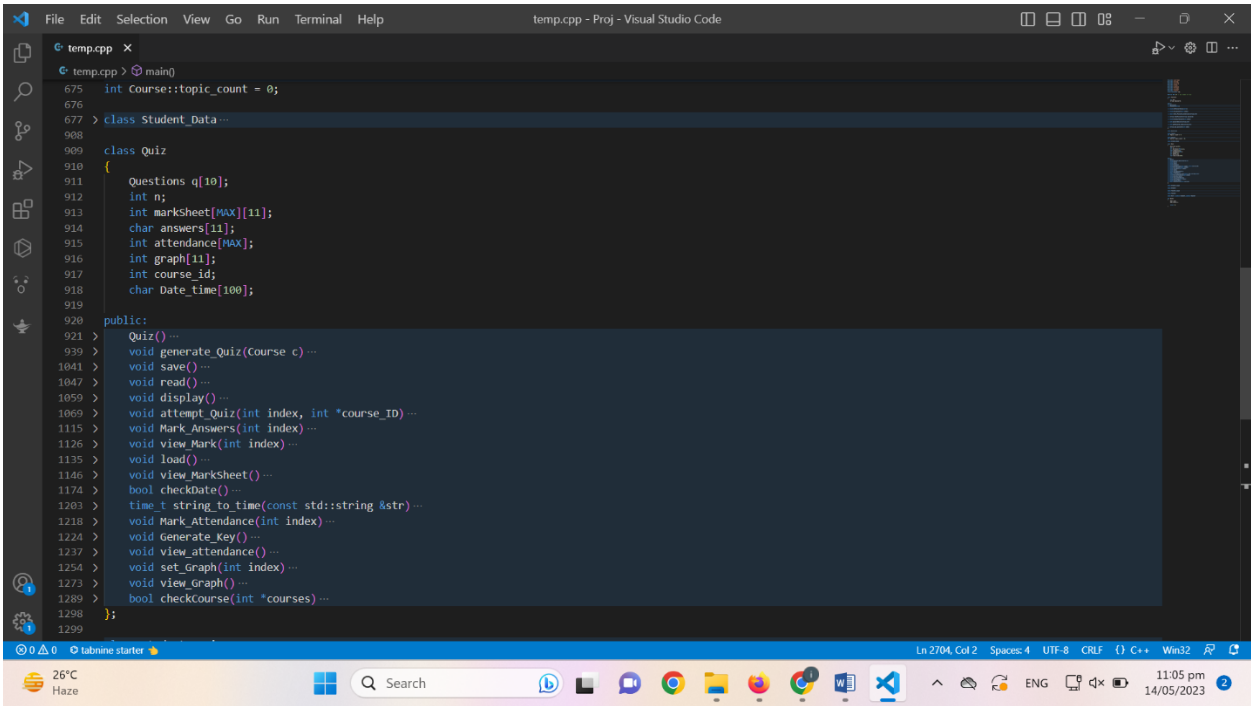
Class Quiz :

This class has a Composition relationship with class Questions. Questions are a part of Quiz.

This is because Quiz class consists of an array of Questions without which It cannot exist. It uses these Question arrays to hold Questions from Courses and make a quiz. Each index contains a Question that was received by returning a Question object from Courses. Courses is not a member of the Quiz class, it is only passed as a Parameter from the Teacher class. This class uses Questions to form quizzes, display and conduct Quizzes along with mark answer, create analytics and other functionalities.

This class also has a Aggregation relationship with class Student_Data class. Quiz has Student_Data.

This is because after conducting the quiz, the class needs to store the marks and attendance of the student in the Student data, this can't be done without the Student_Data class. So Quiz class utilizes the members of the Student_Data class but is not dependent on them to exist to conduct the quizzes.



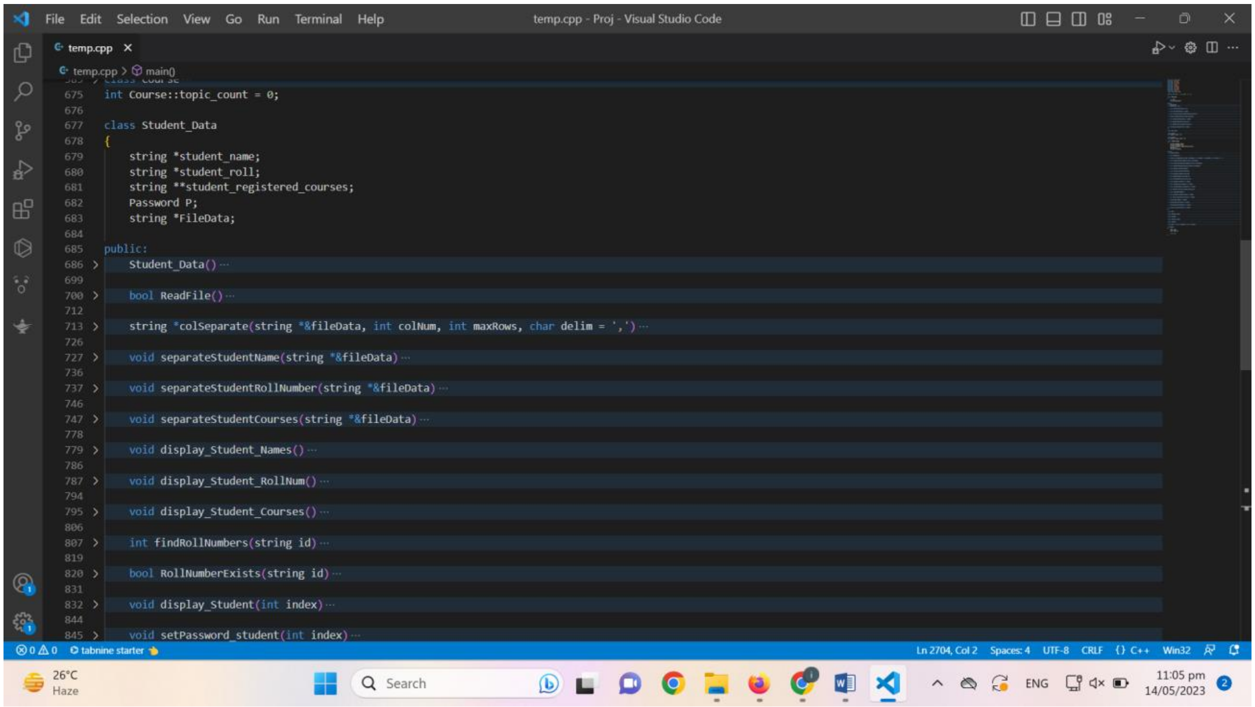
File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
temp.cpp
temp.cpp > main()
Ln 2704, Col 2 Spaces: 4 UTF-8 CRLF {} C++ Win32
tabnine starter
26°C
Haze
Search
ENG
11:05 pm
14/05/2023

```cpp
int Course::topic_count = 0;

class Student_Data ···

class Quiz
{
    Questions q[10];
    int n;
    int markSheet[MAX][11];
    char answers[11];
    int attendance[MAX];
    int graph[11];
    int course_id;
    char Date_time[100];

public:
    Quiz() ···
    void generate_Quiz(Course c) ···
    void save() ···
    void read() ···
    void display() ···
    void attempt_Quiz(int index, int *course_ID) ···
    void Mark_Answers(int index) ···
    void view_Mark(int index) ···
    void load() ···
    void view_MarkSheet() ···
    bool checkDate() ···
    time_t string_to_time(const std::string &str) ···
    void Mark_Attendance(int index) ···
    void Generate_Key() ···
    void view_attendance() ···
    void set_Graph(int index) ···
    void view_Graph() ···
    bool checkCourse(int *courses) ···
};
```

Class Student_Data :

This class has a Composition relationship with class Password. Password is a part of Student_Data

This is because the Student_Data class needs a Password class so that it can store student Ids but the Password can still exist without Student_Data.



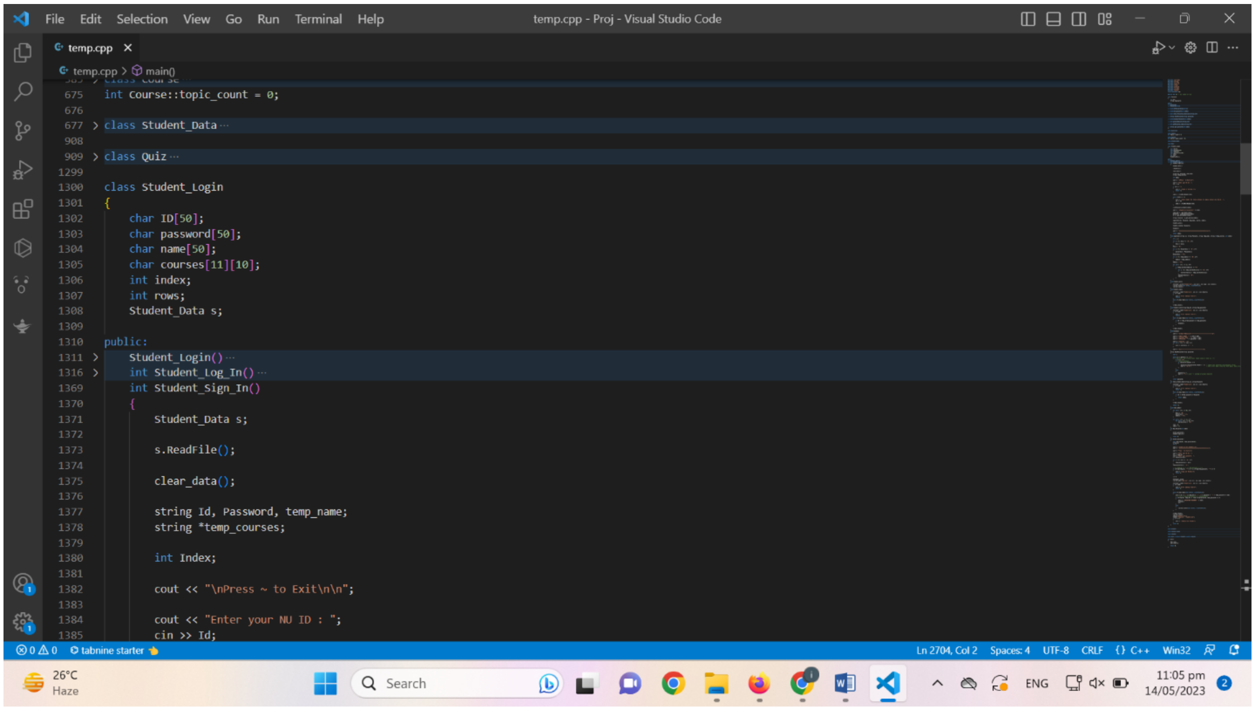
File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
temp.cpp
temp.cpp > main()
Ln 2704, Col 2 Spaces: 4 UTF-8 CRLF {} C++ Win32
tabnine starter
26°C
Haze
Search
ENG
11:05 pm
14/05/2023

```cpp
int Course::topic_count = 0;

class Student_Data
{
    string *student_name;
    string *student_roll;
    string **student_registered_courses;
    Password P;
    string *FileData;

public:
    Student_Data() ···

    bool ReadFile() ···

    string *colSeparate(string *&fileData, int colNum, int maxRows, char delim = ',') ···

    void separateStudentName(string *&fileData) ···

    void separateStudentRollNumber(string *&fileData) ···

    void separateStudentCourses(string *&fileData) ···

    void display_Student_Names() ···

    void display_Student_RollNum() ···

    void display_Student_Courses() ···

    int findRollNumbers(string id) ···

    bool RollNumberExists(string id) ···

    void display_Student(int index) ···

    void setPassword_student(int index) ···
```

Class Student_Login :

This class has a Composition relationship with class Student_Data. Student_Data is a part of Student_Login.

This is because a Student can’t Login without their Data.



```cpp
int Course::topic_count = 0;

class Student_Data ···

class Quiz ···

class Student_Login
{
    char ID[50];
    char password[50];
    char name[50];
    char courses[11][10];
    int index;
    int rows;
    Student_Data s;

public:
    Student_Login() ···
    int Student_Log_In() ···
    int Student_Sign_In()
    {
        Student_Data s;

        s.ReadFile();

        clear_data();

        string Id, Password, temp_name;
        string *temp_courses;

        int Index;

        cout << "\nPress ~ to Exit\n\n";

        cout << "Enter your NU ID : ";
        cin >> Id;
```

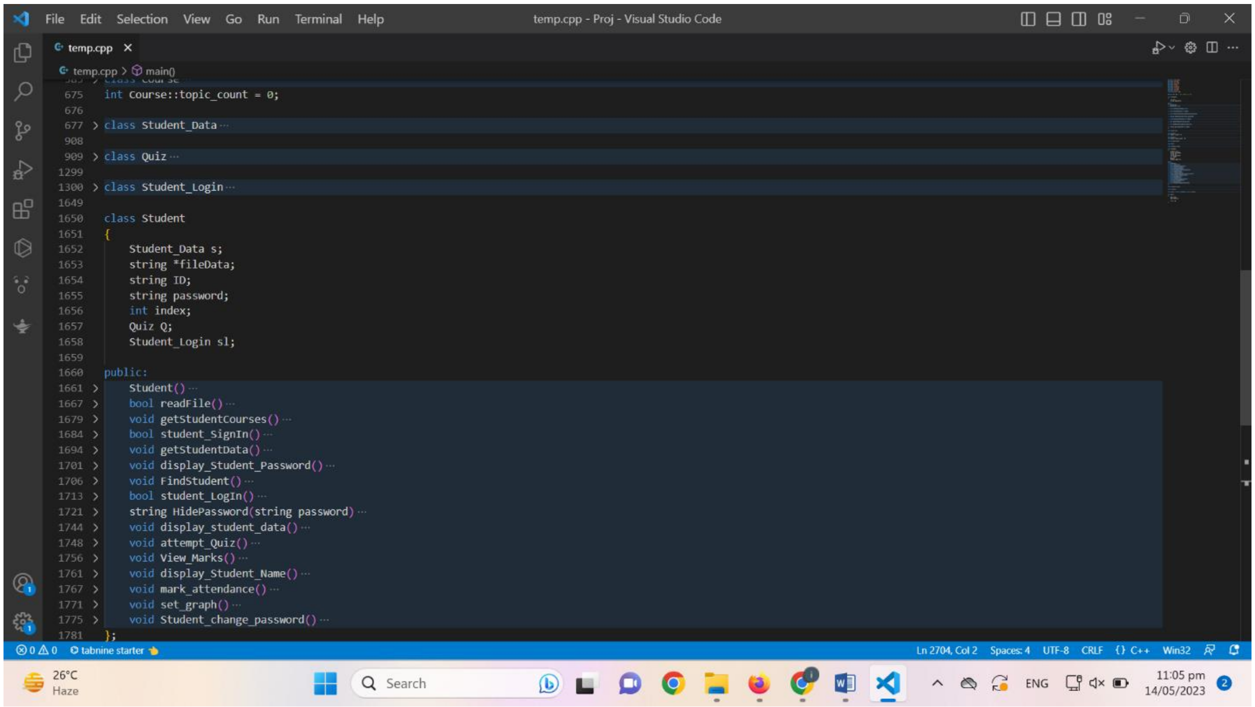
Class Students :

This class has a Composition relationship with classes Student_Data, Student_Login. Student_Data and Student_Login are parts of Students.

This is because a Student can't exist without it's Data. And a Student can't Login without the Student_Login class. That is why, for a student to exist, the class needs data of the student and a way to Login into the Student.

This class has a Aggregation relationship with class Quiz.

This is be Student can exist without giving a Quiz. The Student can also attempt the Quiz too. If there is no Student, there is still a Quiz and if there is no Quiz, there is still a student.



File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
temp.cpp
temp.cpp > main()
0 0 tabnine starter
Ln 2704, Col 2 Spaces: 4 UTF-8 CRLF {} C++ Win32
26°C
Haze
Search
ENG
11:05 pm
14/05/2023

```
int Course::topic_count = 0;

class Student_Data ···

class Quiz ···

class Student_Login ···

class Student
{
    Student_Data s;
    string *fileData;
    string ID;
    string password;
    int index;
    Quiz Q;
    Student_Login sl;

public:
    Student() ···
    bool readFile() ···
    void getStudentCourses() ···
    bool student_SignIn() ···
    void getStudentData() ···
    void display_Student_Password() ···
    void FindStudent() ···
    bool student_LogIn() ···
    string HidePassword(string password) ···
    void display_student_data() ···
    void attempt_Quiz() ···
    void View_Marks() ···
    void display_Student_Name() ···
    void mark_attendance() ···
    void set_graph() ···
    void Student_change_password() ···
};
```

Class Teacher_Login :

It is a class that helps Teachers login or sign in. It maintains teacher's names along with their passwords and deletes them.

Class Teacher :

This class has a Composition relationship with the classes Password, Courses, Teacher_Login.

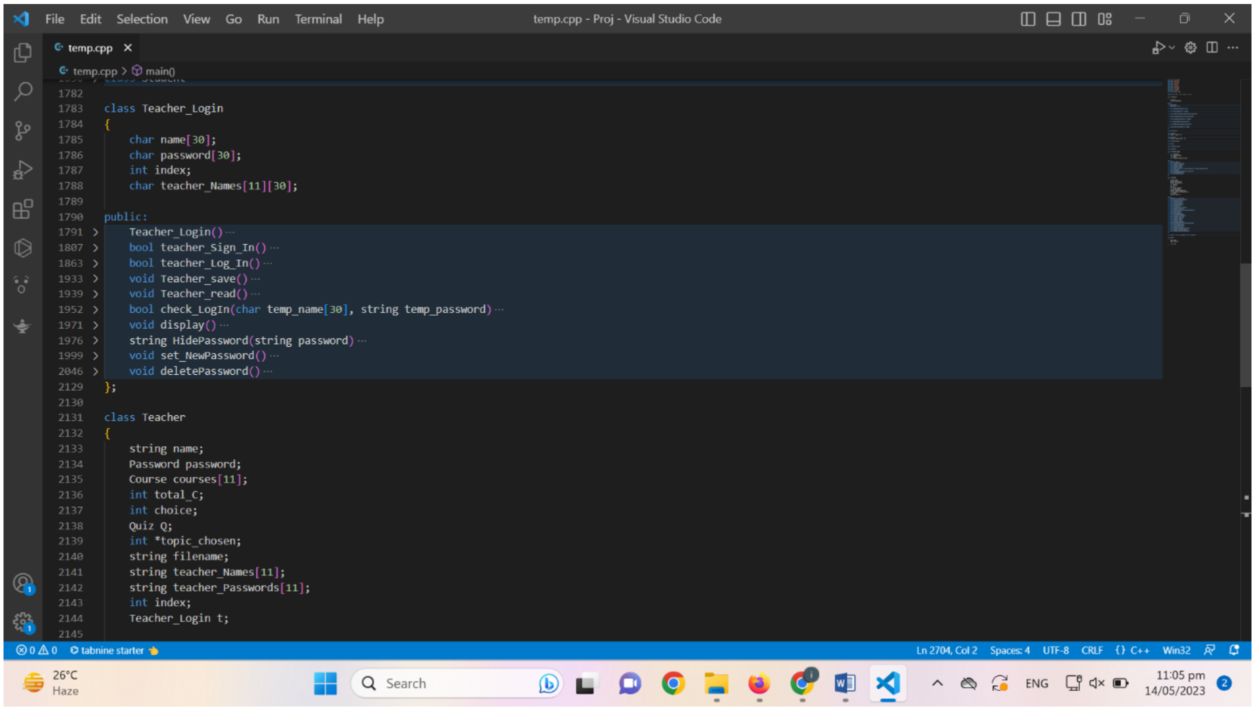
This is because a teacher cannot exist without the teacher having their account data, i.e. their Passwords. Logging in is a part of what the Teacher does. Teacher needs their password to form an account that can be saved.

A teacher also needs a course. A course is a part of the Teacher's identity and work. We often associate our teachers with their courses. Without a course, a teacher isn't a teacher.

And the Teacher class needs the Teacher_Login class to be able to log into their account or else there wont be any teacher. Logging in / Signing in is a part of Teacher.

This class has an Aggregation relationship with the class Quiz.

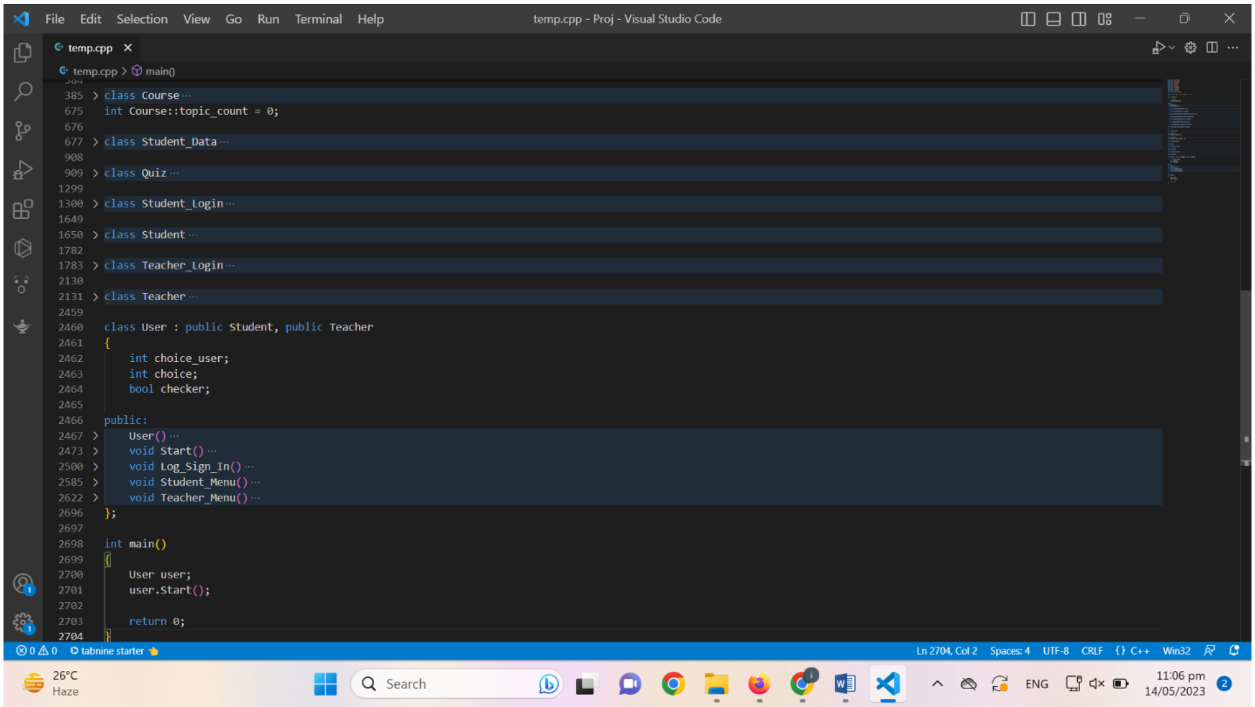
This is because a teacher conducts the quizzes. But without quizzes, a teacher is still a teacher.


File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
temp.cpp
temp.cpp > main()
Ln 2704, Col 2 Spaces: 4 UTF-8 CRLF {} C++ Win32
tabnine starter
26°C Haze
Search
ENG
11:05 pm 14/05/2023

```cpp
class Teacher_Login
{
    char name[30];
    char password[30];
    int index;
    char teacher_Names[11][30];

public:
    Teacher_Login() ···
    bool teacher_Sign_In() ···
    bool teacher_Log_In() ···
    void Teacher_save() ···
    void Teacher_read() ···
    bool check_LogIn(char temp_name[30], string temp_password) ···
    void display() ···
    string HidePassword(string password) ···
    void set_NewPassword() ···
    void deletePassword() ···
};

class Teacher
{
    string name;
    Password password;
    Course courses[11];
    int total_C;
    int choice;
    Quiz Q;
    int *topic_chosen;
    string filename;
    string teacher_Names[11];
    string teacher_Passwords[11];
    int index;
    Teacher_Login t;
```

Class User :

A Student is a User. A teacher is a User. So both Student and Teacher class are inherited into the User class. This class manages both Student and Teacher classes's functionalities and contains their menus.


File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
temp.cpp
temp.cpp > main()
Ln 2704, Col 2 Spaces: 4 UTF-8 CRLF {} C++ Win32
tabnine starter
26°C Haze
Search
ENG
11:06 pm 14/05/2023

```cpp
class Course ···
int Course::topic_count = 0;

class Student_Data ···

class Quiz ···

class Student_Login ···

class Student ···

class Teacher_Login ···

class Teacher ···

class User : public Student, public Teacher
{
    int choice_user;
    int choice;
    bool checker;

public:
    User() ···
    void Start() ···
    void Log_Sign_In() ···
    void Student_Menu() ···
    void Teacher_Menu() ···
};

int main()
{
    User user;
    user.Start();

    return 0;
}
```

HOW TO USE THE EXAMINIATION SYSTEM

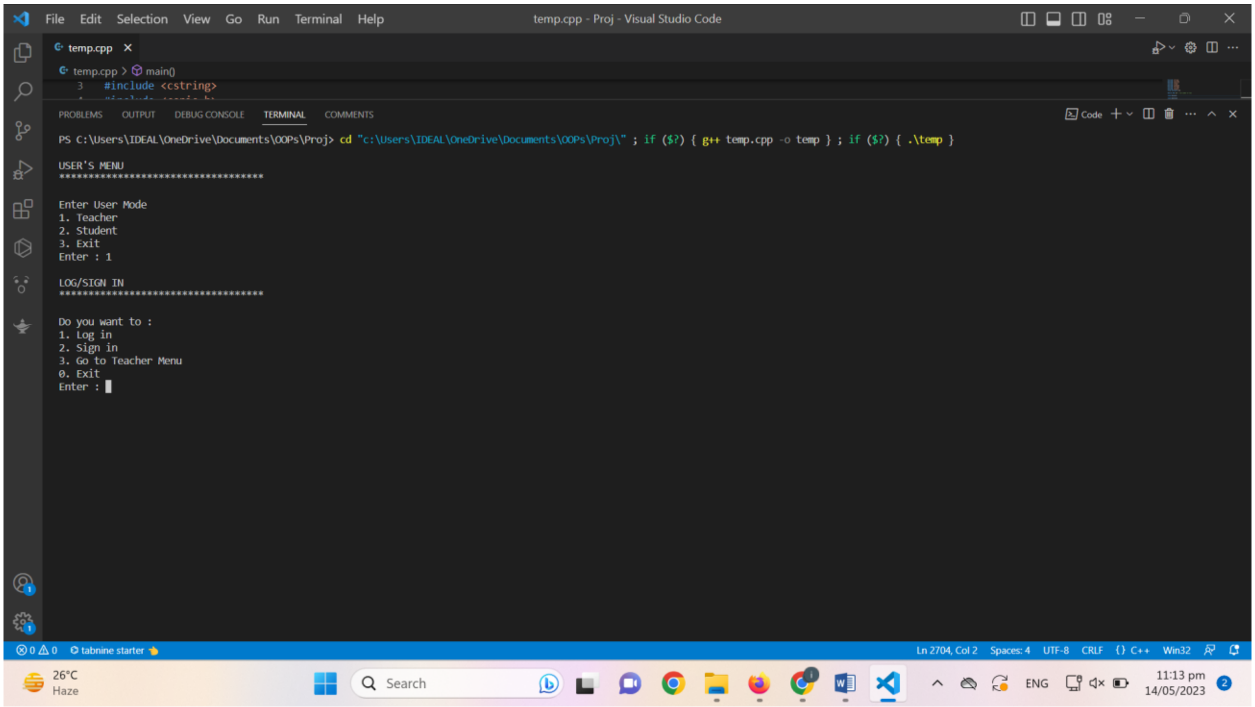
There are 2 ways of using the examination system. It is targeted towards 2 different users.

1. Teacher
2. Student

Lets talk first about how will the Teacher use the system.

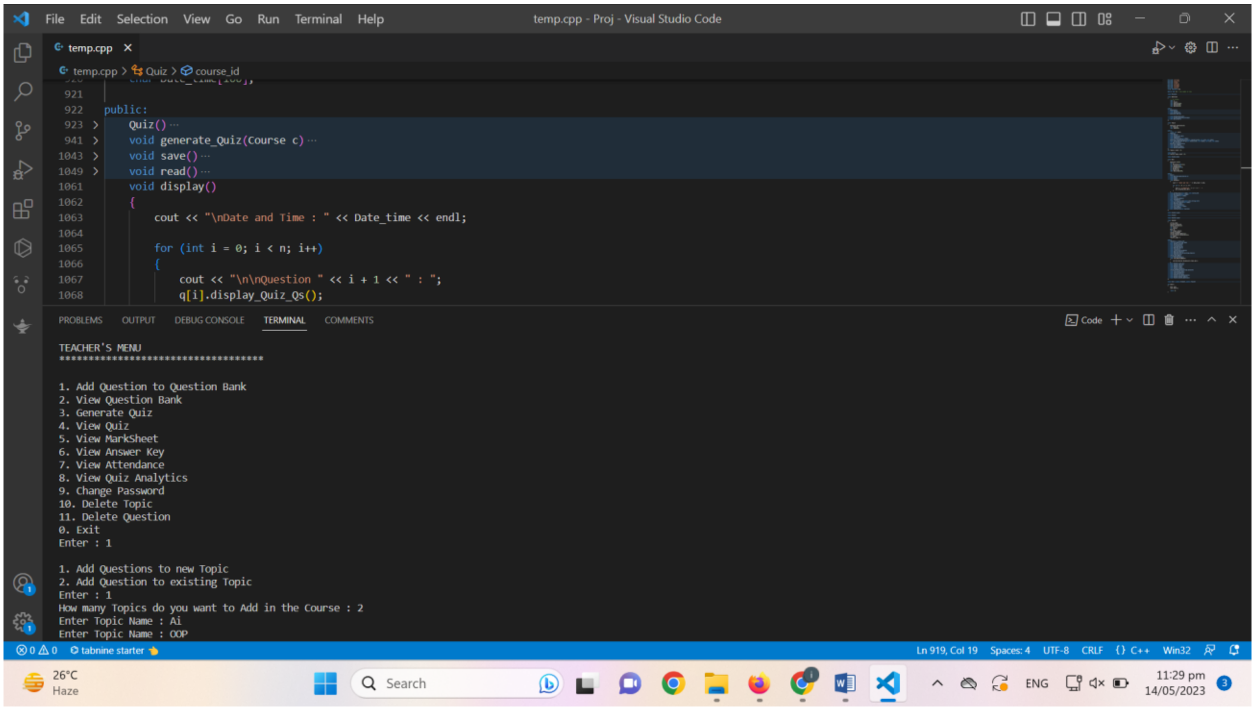
Teacher :

1. Run the Program.
2. User Menu will appear, Press 1 to enter Teacher Mode or 3 to exit the program.
3. Once in Teacher Mode, you can either sign in if you haven't already done that before. Or log in if you are already signed in. During logging/signing in, if you want to exit this part and go back to the Log/Sign in Menu, Enter ~ for name and password. In this menu, you can also exit the program.
4. Once you are logged/signed in, Press 3 to choose your course. Here all courses will pop up and you can choose which course you want to conduct a Quiz in.
5. Then enter a course number to open up Teacher Menu. Here you will have many options to choose from. You can press 1 to add Questions to a Question Bank, then deciding if you want to add a question to an existing topic or a new one.
6. Press 2 to view the Question Bank.
7. Press 3 to Create a Quiz from the Question Bank, here you are given 3 options. Option 1 allows you to choose the topic but randomizes the Questions. Option 2 allows the teacher to choose the topic and the questions are in order. Option 3 completely randomizes the quiz where even the topics are chosen at random by the system along with the questions. The answer key if also stored away. After the Questions have been chosen, the teacher is required to enter the Date before which the quiz must be attempted, after which, no one can attempt the quiz. The input of date and time must be according to the given example.
8. You can now view the quiz, marksheet of students who have attempted the quiz, answer key, attendance of the students of have attempted the quiz, quiz analytics, change your password, delete a topic or/and a question by pressing the corresponding numbers and entering the correct and desired inputs.
9. Once you have done everything, press 0 to exit.


File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
temp.cpp
temp.cpp > main()
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
Ln 2704, Col 2 Spaces: 4 UTF-8 CRLF {} C++ Win32
tabnine starter
26°C Haze
Search
ENG
11:13 pm 14/05/2023

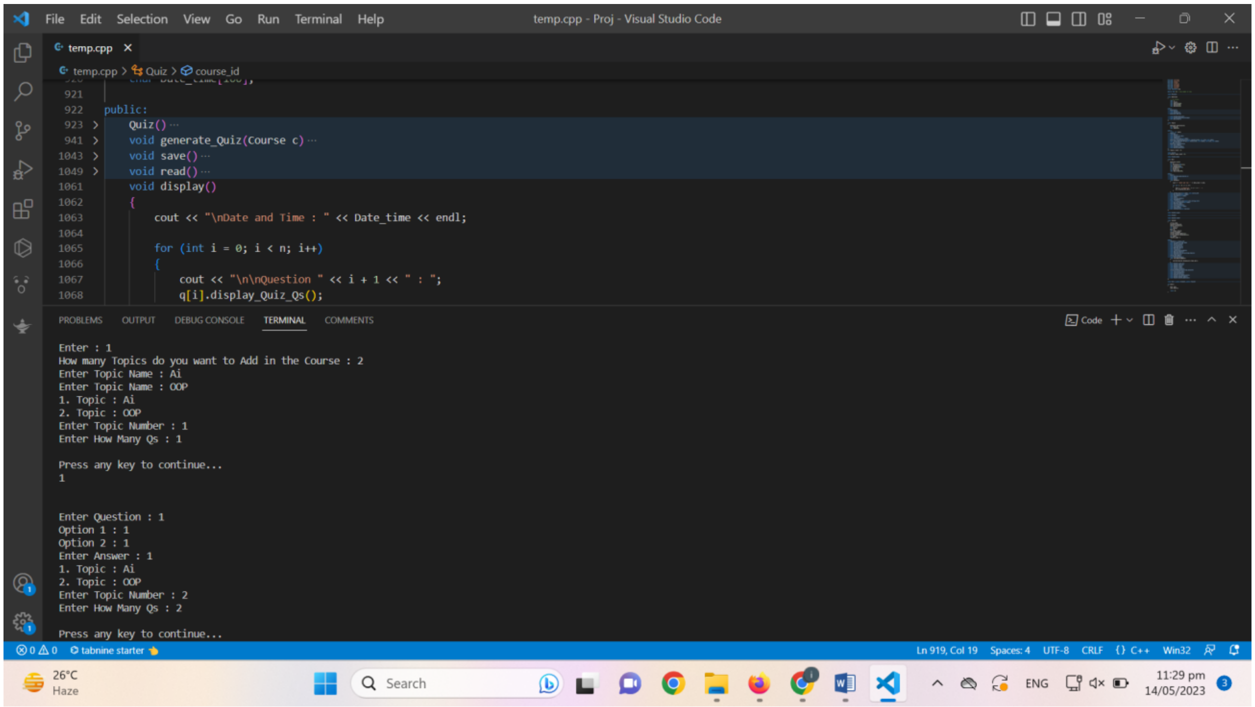
```cpp
#include <cstring>
```

```
PS C:\Users\IDEAL\OneDrive\Documents\OOPs\Proj> cd "c:\Users\IDEAL\OneDrive\Documents\OOPs\Proj\" ; if ($?) { g++ temp.cpp -o temp } ; if ($?) { .\temp }

USER'S MENU
***********************************

Enter User Mode
1. Teacher
2. Student
3. Exit
Enter : 1

LOG/SIGN IN
***********************************

Do you want to :
1. Log in
2. Sign in
3. Go to Teacher Menu
0. Exit
Enter : 
```


File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
temp.cpp
temp.cpp > Quiz > course_id
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
Ln 919, Col 19 Spaces: 4 UTF-8 CRLF {} C++ Win32
tabnine starter
26°C Haze
Search
ENG
11:29 pm 14/05/2023

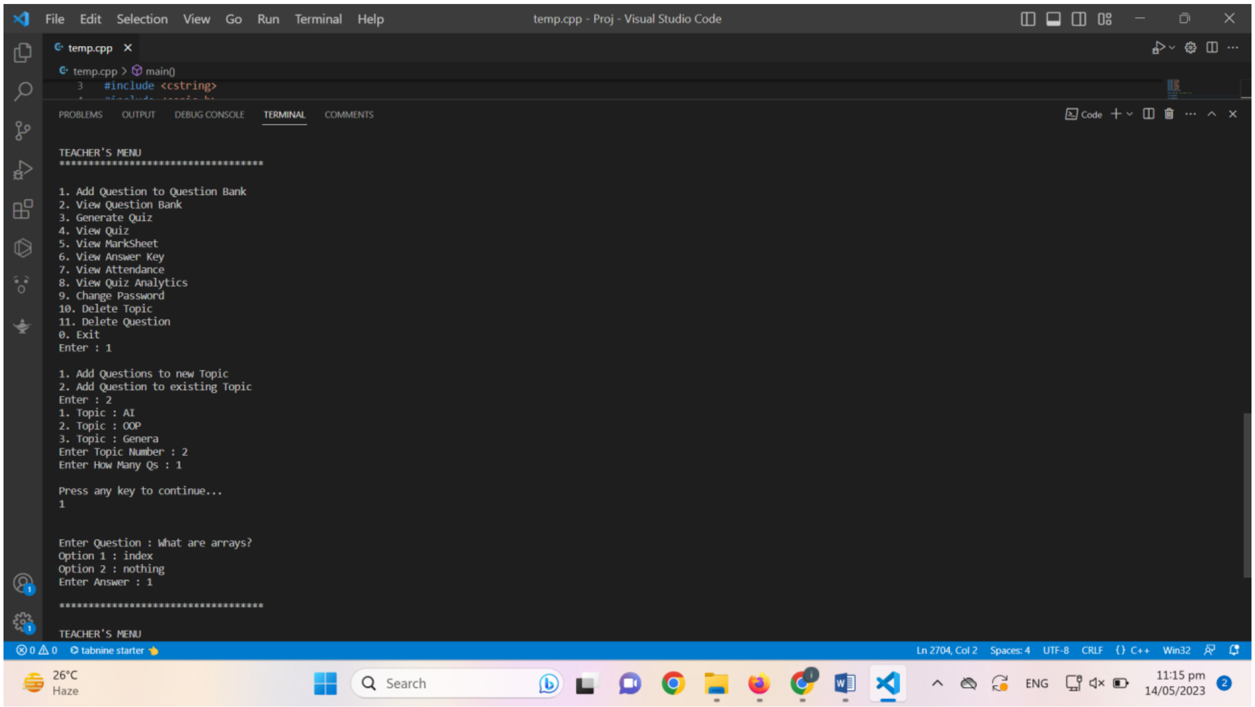
```cpp

public:
    Quiz() ···
    void generate_Quiz(Course c) ···
    void save() ···
    void read() ···
    void display()
    {
        cout << "\nDate and Time : " << Date_time << endl;

        for (int i = 0; i < n; i++)
        {
            cout << "\n\nQuestion " << i + 1 << " : ";
            q[i].display_Quiz_Qs();
```

```
TEACHER'S MENU
***********************************

1. Add Question to Question Bank
2. View Question Bank
3. Generate Quiz
4. View Quiz
5. View MarkSheet
6. View Answer Key
7. View Attendance
8. View Quiz Analytics
9. Change Password
10. Delete Topic
11. Delete Question
0. Exit
Enter : 1

1. Add Questions to new Topic
2. Add Question to existing Topic
Enter : 1
How many Topics do you want to Add in the Course : 2
Enter Topic Name : Ai
Enter Topic Name : OOP
```


File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
temp.cpp
temp.cpp > Quiz > course_id
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
Code
0 0 tabnine starter
Ln 919, Col 19 Spaces: 4 UTF-8 CRLF C++ Win32
26°C
Haze
Search
ENG
11:29 pm
14/05/2023

```cpp

public:
    Quiz() ···
    void generate_Quiz(Course c) ···
    void save() ···
    void read() ···
    void display()
    {
        cout << "\nDate and Time : " << Date_time << endl;

        for (int i = 0; i < n; i++)
        {
            cout << "\n\nQuestion " << i + 1 << " : ";
            q[i].display_Quiz_Qs();
```

```
Enter : 1
How many Topics do you want to Add in the Course : 2
Enter Topic Name : Ai
Enter Topic Name : OOP
1. Topic : Ai
2. Topic : OOP
Enter Topic Number : 1
Enter How Many Qs : 1

Press any key to continue...
1


Enter Question : 1
Option 1 : 1
Option 2 : 1
Enter Answer : 1
1. Topic : Ai
2. Topic : OOP
Enter Topic Number : 2
Enter How Many Qs : 2

Press any key to continue...
```

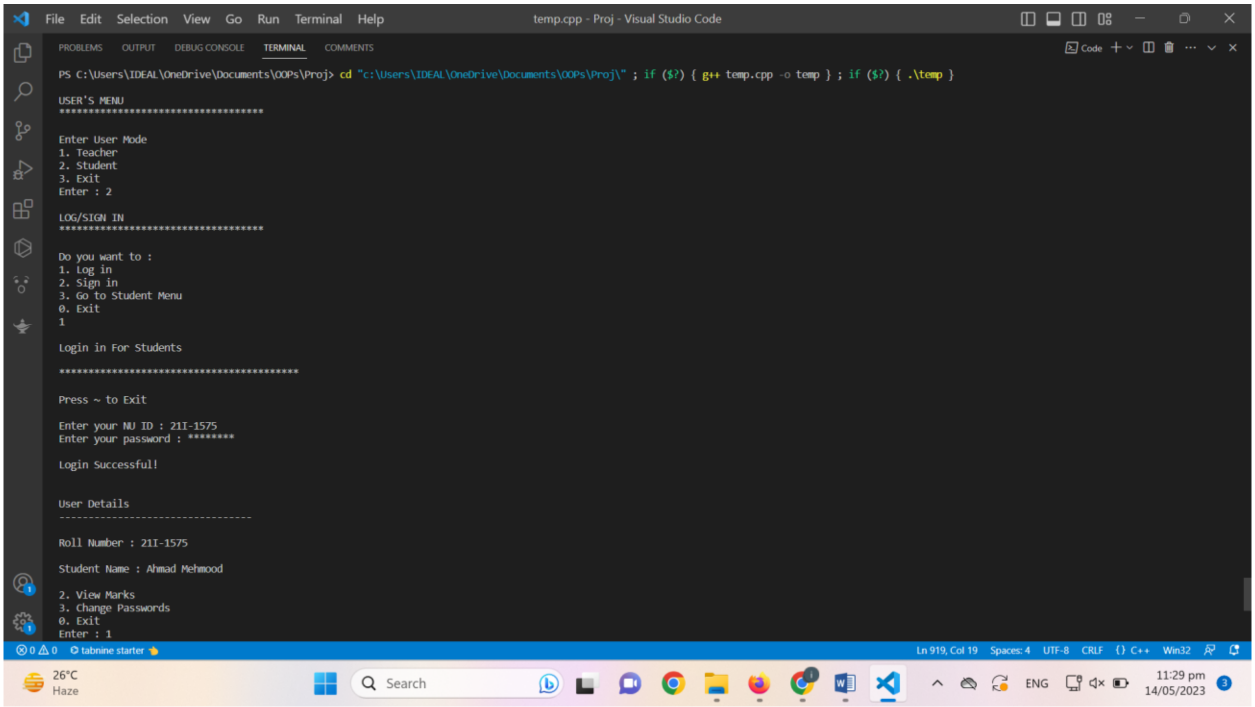

File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
temp.cpp
temp.cpp > main()
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
Code
0 0 tabnine starter
Ln 2704, Col 2 Spaces: 4 UTF-8 CRLF C++ Win32
26°C
Haze
Search
ENG
11:15 pm
14/05/2023

```cpp
#include <cstring>
```

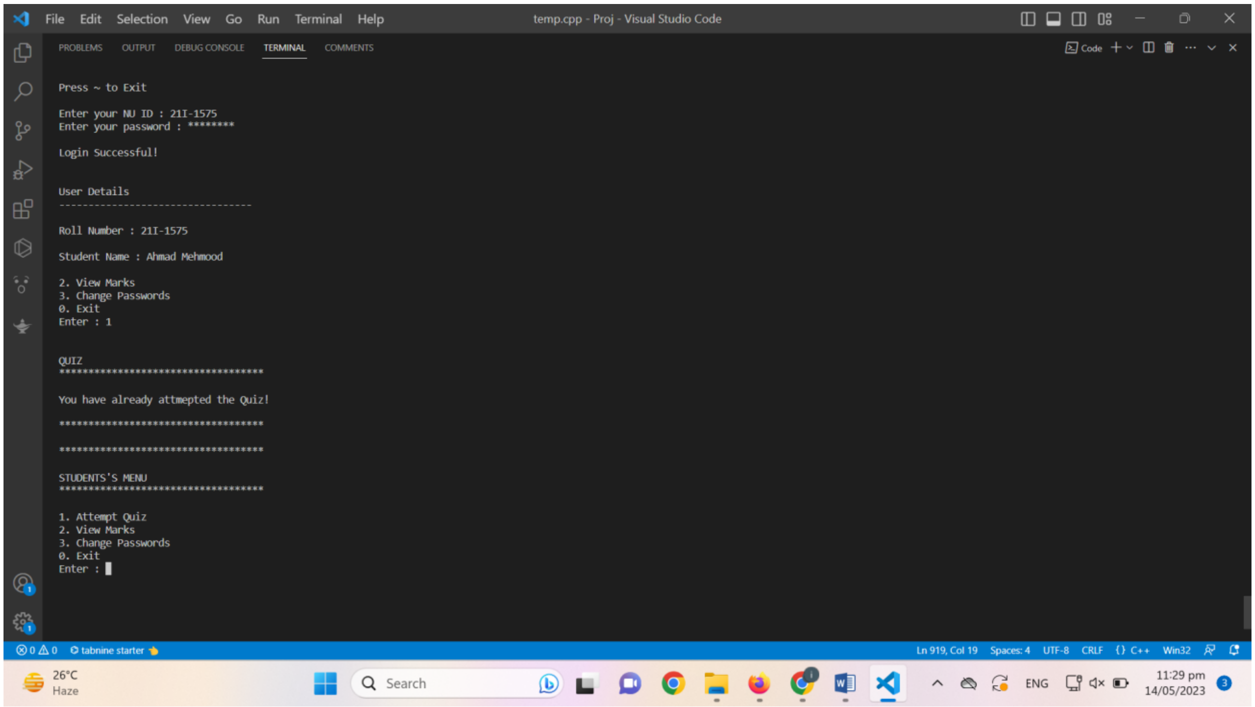
```
TEACHER'S MENU
***********************************

1. Add Question to Question Bank
2. View Question Bank
3. Generate Quiz
4. View Quiz
5. View MarkSheet
6. View Answer Key
7. View Attendance
8. View Quiz Analytics
9. Change Password
10. Delete Topic
11. Delete Question
0. Exit
Enter : 1

1. Add Questions to new Topic
2. Add Question to existing Topic
Enter : 2
1. Topic : AI
2. Topic : OOP
3. Topic : Genera
Enter Topic Number : 2
Enter How Many Qs : 1

Press any key to continue...
1


Enter Question : What are arrays?
Option 1 : index
Option 2 : nothing
Enter Answer : 1

***********************************

TEACHER'S MENU
```

Student :

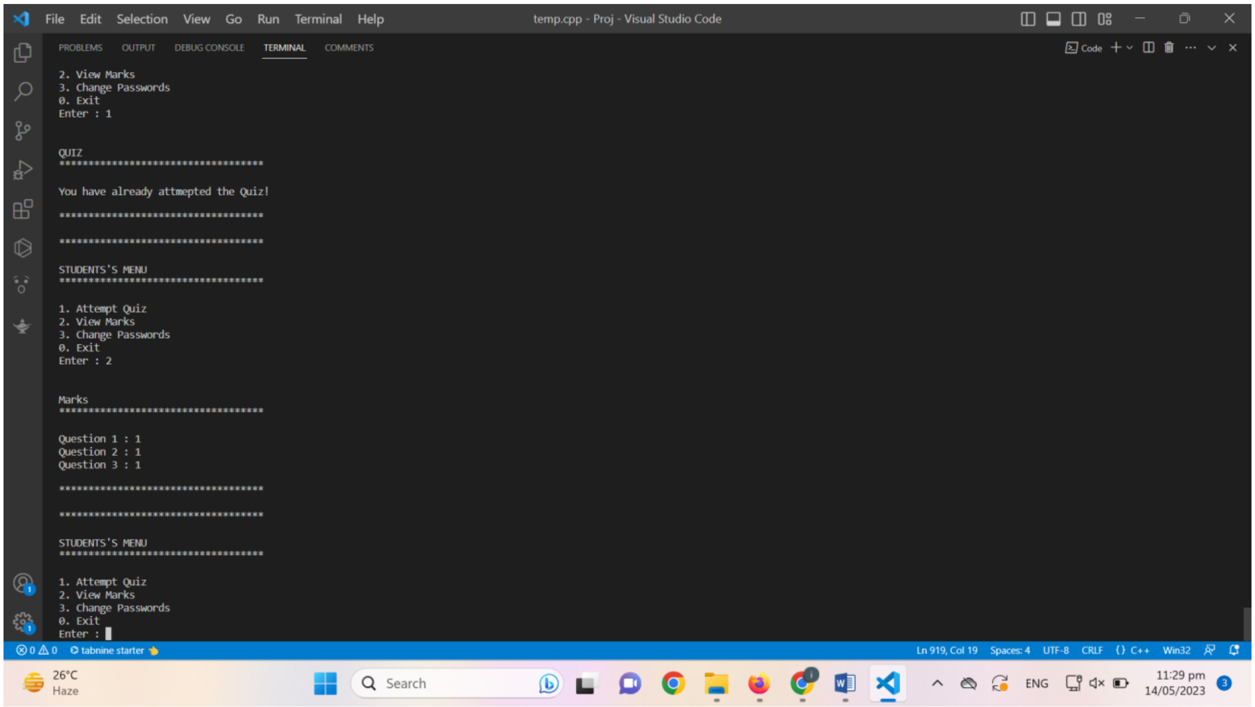
1. Run the Program.
2. User Menu will appear, Press 2 to enter Student Mode or 3 to exit the program.
3. Once in Student Mode, you can either sign in if you haven't already done that before. Or log in if you are already signed in. During logging/signing in, if you want to exit this part and go back to the Log/Sign in Menu, Enter ~ for name and password. In this menu, you can also exit the program.
4. Once you have logged/signed in, press 3 to enter Student Menu, this menu will allow the student to perform 3 activities.
5. Press 1 to attempt the Quiz. If the user is registered in the courses and tries to attempt the quiz before the deadline, the the quiz Questions will pop up and the user will be asked to enter the correct options.
6. Once all options are inputted, user is returned back to the Student Menu where they can press 2 to view their marks. They also wont be able to attempt the same quiz twice.
7. User can press 3 to change their password too. First Enter you Current ID and Password, then you will be asked to enter your new ID and Password.
8. Once you have done everything, press 0 to exit.


File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
Ln 919, Col 19 Spaces: 4 UTF-8 CRLF {} C++ Win32
tabnine starter
26°C Haze
Search
ENG
11:29 pm 14/05/2023

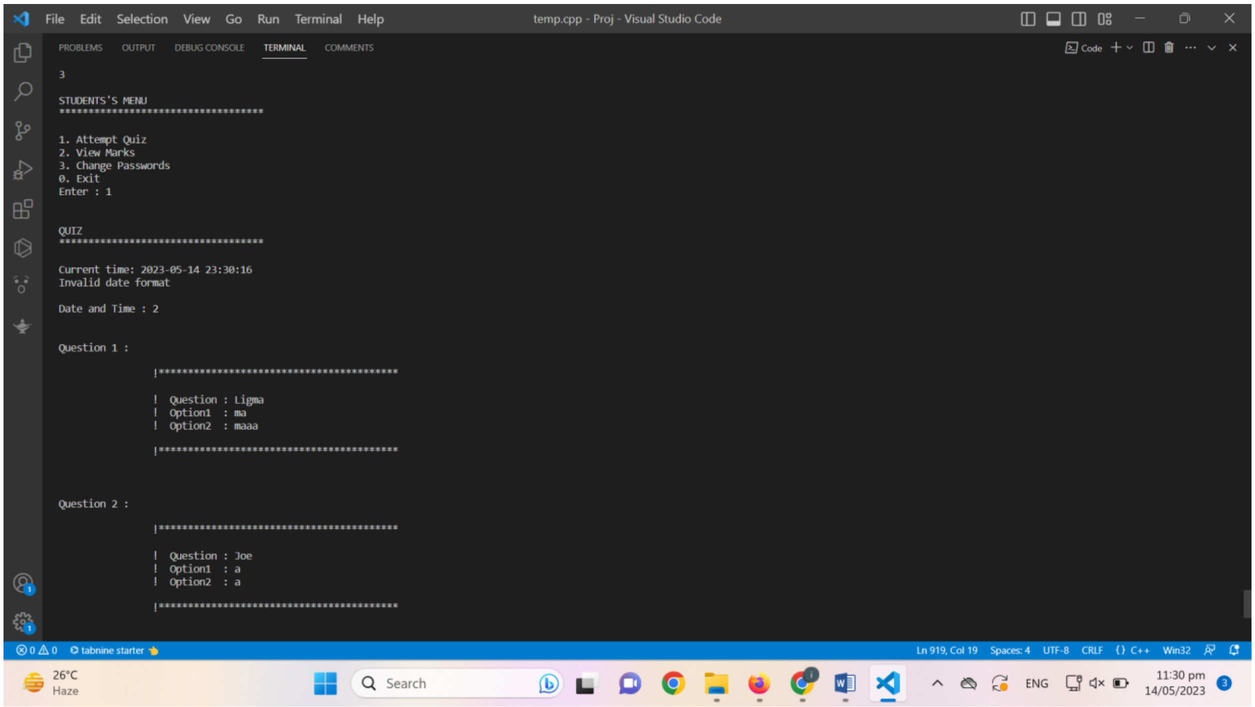
```
PS C:\Users\IDEAL\OneDrive\Documents\OOPs\Proj> cd "c:\Users\IDEAL\OneDrive\Documents\OOPs\Proj\" ; if ($?) { g++ temp.cpp -o temp } ; if ($?) { .\temp }

USER'S MENU
***********************************

Enter User Mode
1. Teacher
2. Student
3. Exit
Enter : 2

LOG/SIGN IN
***********************************

Do you want to :
1. Log in
2. Sign in
3. Go to Student Menu
0. Exit
1

Login in For Students

*****************************************

Press ~ to Exit

Enter your NU ID : 21I-1575
Enter your password : ********

Login Successful!

User Details
---------------------------------

Roll Number : 21I-1575

Student Name : Ahmad Mehmood

2. View Marks
3. Change Passwords
0. Exit
Enter : 1
```


File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
Ln 919, Col 19 Spaces: 4 UTF-8 CRLF {} C++ Win32
tabnine starter
26°C Haze
Search
ENG
11:29 pm 14/05/2023

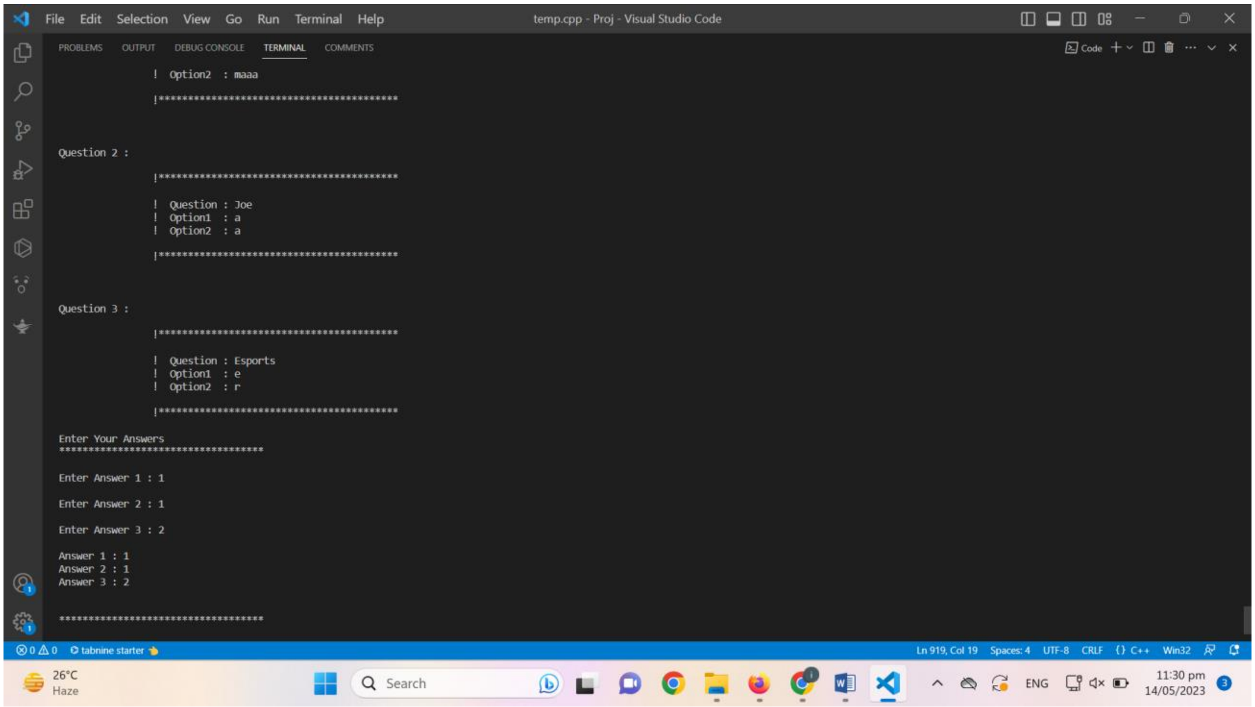
```
Press ~ to Exit

Enter your NU ID : 21I-1575
Enter your password : ********

Login Successful!

User Details
---------------------------------

Roll Number : 21I-1575

Student Name : Ahmad Mehmood

2. View Marks
3. Change Passwords
0. Exit
Enter : 1


QUIZ
***********************************

You have already attmepted the Quiz!

***********************************

***********************************

STUDENTS'S MENU
***********************************

1. Attempt Quiz
2. View Marks
3. Change Passwords
0. Exit
Enter : 
```


File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
Code
0 0 tabnine starter
Ln 919, Col 19 Spaces: 4 UTF-8 CRLF {} C++ Win32
26°C Haze
Search
ENG
11:29 pm 14/05/2023

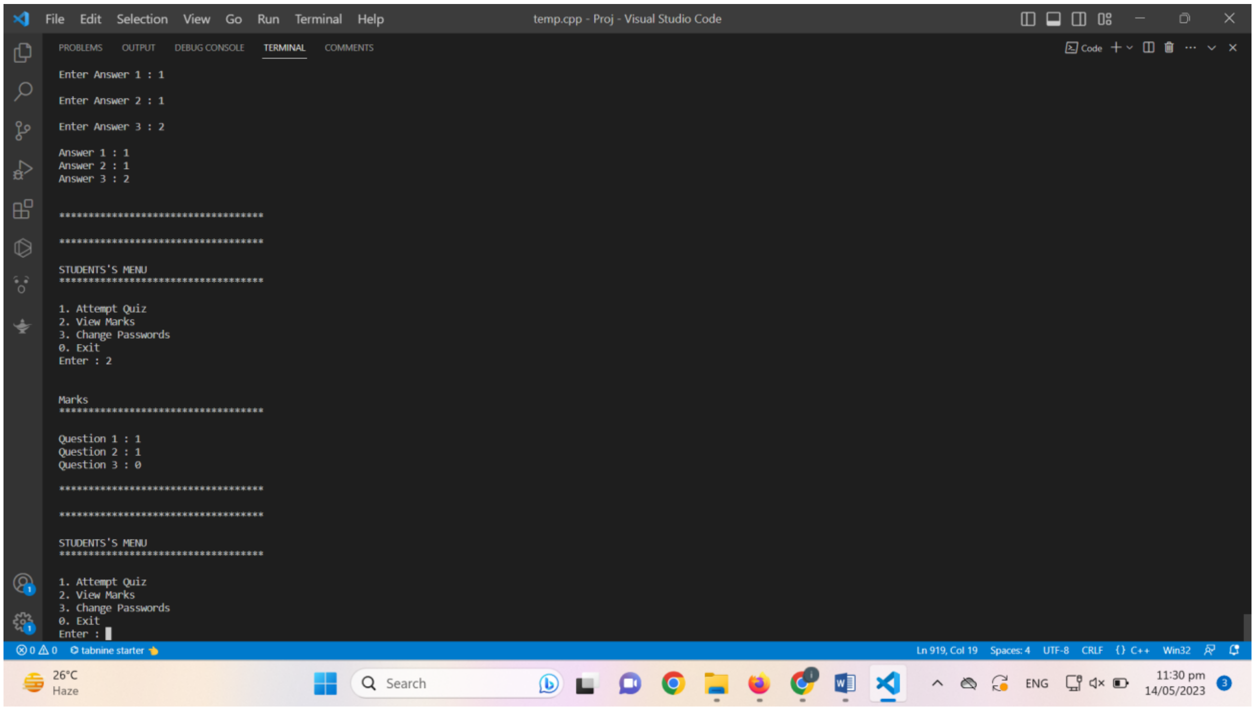
```
2. View Marks
3. Change Passwords
0. Exit
Enter : 1


QUIZ
***********************************

You have already attmepted the Quiz!

***********************************

***********************************

STUDENTS'S MENU
***********************************

1. Attempt Quiz
2. View Marks
3. Change Passwords
0. Exit
Enter : 2


Marks
***********************************

Question 1 : 1
Question 2 : 1
Question 3 : 1

***********************************

***********************************

STUDENTS'S MENU
***********************************

1. Attempt Quiz
2. View Marks
3. Change Passwords
0. Exit
Enter : 
```


File Edit Selection View Go Run Terminal Help
temp.cpp - Proj - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS
Code
0 0 tabnine starter
Ln 919, Col 19 Spaces: 4 UTF-8 CRLF {} C++ Win32
26°C Haze
Search
ENG
11:30 pm 14/05/2023

```
3

STUDENTS'S MENU
***********************************

1. Attempt Quiz
2. View Marks
3. Change Passwords
0. Exit
Enter : 1


QUIZ
***********************************

Current time: 2023-05-14 23:30:16
Invalid date format

Date and Time : 2


Question 1 :

                |****************************************

                !  Question : Ligma
                !  Option1  : ma
                !  Option2  : maaa

                |****************************************



Question 2 :

                |****************************************

                !  Question : Joe
                !  Option1  : a
                !  Option2  : a

                |****************************************
```



```
                !  Option2  : maaa

                |****************************************



Question 2 :

                |****************************************

                !  Question : Joe
                !  Option1  : a
                !  Option2  : a

                |****************************************



Question 3 :

                |****************************************

                !  Question : Esports
                !  Option1  : e
                !  Option2  : r

                |****************************************

Enter Your Answers
***********************************

Enter Answer 1 : 1

Enter Answer 2 : 1

Enter Answer 3 : 2

Answer 1 : 1
Answer 2 : 1
Answer 3 : 2


***********************************
```



```
Enter Answer 1 : 1

Enter Answer 2 : 1

Enter Answer 3 : 2

Answer 1 : 1
Answer 2 : 1
Answer 3 : 2


***********************************

***********************************

STUDENTS'S MENU
***********************************

1. Attempt Quiz
2. View Marks
3. Change Passwords
0. Exit
Enter : 2


Marks
***********************************

Question 1 : 1
Question 2 : 1
Question 3 : 0

***********************************

***********************************

STUDENTS'S MENU
***********************************

1. Attempt Quiz
2. View Marks
3. Change Passwords
0. Exit
Enter : 
```