**Assignment on C++ Structure**

**Submitted by Avinash Singh (S211165200104)**

**-------------------------------------------------------------------------------**

**1.**Give the output of the following program. Assuming all the desired header files are already included, which are required to run the code.

struct Pixel  
{  
            int C, R;  
};

void Display(Pixel P)  
{  
            cout << "Col "<< P.C << " Row " << P.R << endl;  
}

int main()  
{            Pixel X = {40,50}, Y, Z;  
            Z = X;  
            X.C += 10;  
            Y = Z;  
            Y.C += 10;  
            Y.R += 20;  
            Z.C -= 15;  
            Display(X);  
            Display(Y);  
            Display(Z);

            return 0;  
}

Give the answer.

Col 50 Row 50

Col 50 Row 70

Col 25 Row 50

**2.**Find the output of the following program. Assuming all the desired header files are already included, which are required to run the code.

struct Play  
{  
            int score, bonus;  
};

void calculate(Play &P, int N = 10)  
{  
            P.score++;  
            P.bonus += N;  
}

int main()  
{  
            Play PL = {10, 15};  
            calculate(PL, 5);  
            cout << PL.score << ":" << PL.bonus << endl;  
            calculate(PL);  
            cout << PL.score << ":" << PL.bonus << endl;  
            calculate(PL, 15);  
            cout << PL.score << ":" << PL.bonus << endl;

            return 0;  
}

Give the answer.

11:20

12:30

13:45

**3.**Find the output of the following program. Assuming all the desired header files are already included, which are required to run the code.

struct MyBox  
{  
            int length, breadth, height;  
};

void dimension (MyBox M)  
{  
            cout << M.length << "x" << M.breadth << "x";  
            cout << M.height << endl;  
}

int main ()  
{  
            MyBox B1 = {10, 15, 5}, B2, B3;  
            ++B1.height;  
            dimension(B1);  
            B3 = B1;  
            ++B3.length;  
            B3.breadth++;  
            dimension(B3);  
            B2 = B3;  
            B2.height += 5;  
            B2.length--;  
            dimension(B2);

           return 0;  
}

Give the answer.

10x15x6

11x16x6

10x16x11

**4.**Rewrite the following program after removing the syntactical errors (if any). Underline each correction.  
  
struct Pixels  
{  
            int color, style;  
}

void showPoint(Pixels P)  
{  
            cout << P.color, P.style << endl;  
}

int main()  
{  
            Pixels Point1 = (5, 3);  
            showPoint(Point1);  
            Pixels Point2 = Point1;  
            color.Point1 += 2;  
            showPoint(Point2);

            return 0;  
}

ans:

/\* after Correcting Program

#include<iostream>

using namespace std;

struct Pixels

{

int color, style;

};

void showPoint(Pixels P)

{

cout << P.color<<","<< P.style << endl;

}

int main()

{

Pixels Point1 = {5, 3};

showPoint(Point1);

Pixels Point2 = Point1;

Point1.color += 2; //if Point2.color += 2;

showPoint(Point2);

return 0;

}

\*/

//if Point1.color+=2;

5,3

5,3

// Point2.color += 2;

5,3

7,3

**5.**Declare a structure to represent a complex number (a number having a real part and imaginary part). Write C++ functions to add, subtract, multiply and divide two complex numbers.

Ans:

//Including all the required header files

#include<iostream>

using namespace std;

//Creating a stucture to that contains real and imginary variables

struct ComplexArith

{

            int real, imginary;

};

void CalcCA(int ch, ComplexArith cn1, ComplexArith cn2)

{

    //Creating object for structure ComplexArith to store value

    struct ComplexArith ans={0,0};

    //using switch to perform calculation

    switch(ch)

    {

        case 1 :

        {

            ans.real=cn1.real+cn2.real; ans.imginary=cn1.imginary+cn2.imginary;

            cout<<"After Addition the solution is "<<ans.real<<"+"<<ans.imginary<<"i"<<endl;

        }break;

         case 2 :

        {

            ans.real=cn1.real-cn2.real; ans.imginary=cn1.imginary-cn2.imginary;

            cout<<"After Subtraction the solution is "<<ans.real<<"+"<<ans.imginary<<"i"<<endl;

        }break;

         case 3 :

        {

            ans.real=cn1.real\*cn2.real; ans.imginary=cn1.imginary\*cn2.imginary;

            cout<<"After Addition the solution is "<<ans.real<<"+"<<ans.imginary<<"i"<<endl;

        }break;

         case 4 :

        {

            ans.real=cn1.real/cn2.real; ans.imginary=cn1.imginary/cn2.imginary;

            cout<<"After Addition the solution is "<<ans.real<<"+"<<ans.imginary<<"i"<<endl;

        }break;

    }

}

int main()

{

            int choice;

            //Declaring 3 object for struct ComplexArith

            struct ComplexArith ca[2];

            //Input of values of two complex numbers

            cout<<"Enter real and imaginary values of input 1 : ";

            cin>>ca[0].real>>ca[0].imginary;

            cout<<"Enter real and imaginary values of input 2 : ";

            cin>>ca[1].real>>ca[1].imginary;

            //Displaying content to perform arithematic operation and take input from user

            cout<<"Enter 1 for performing addition.\nEnter 2 for performing subtration\n";

            cout<<"Enter 3 for performing multiplication\nEnter 4 for performing division";

            cout<<"\nEnter your choice : ";

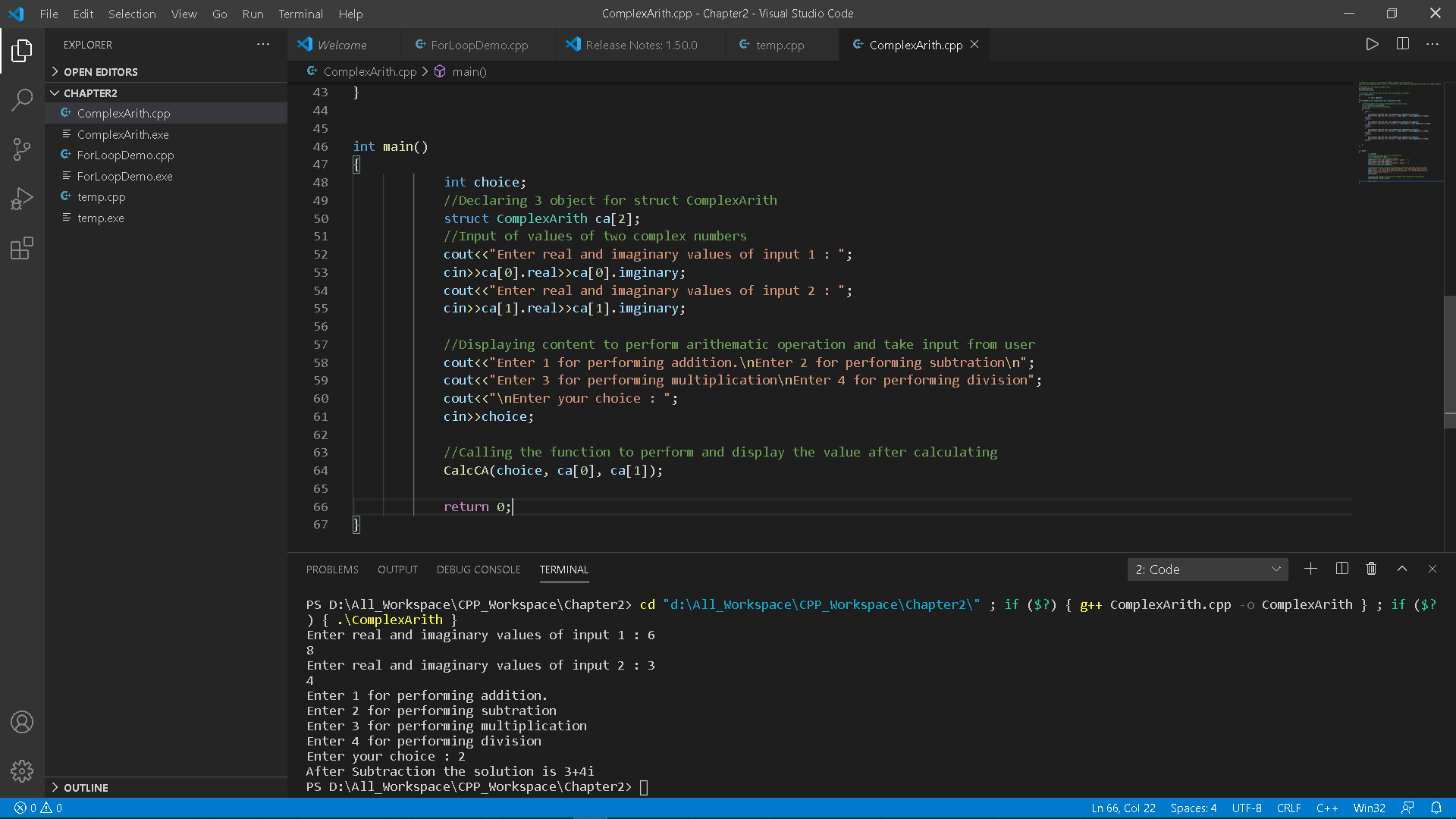
            cin>>choice;

            //Calling the function to perform and display the value after calculating

            CalcCA(choice, ca[0], ca[1]);

            return 0;

}

Output:

**6.**An array stores details of 25 students (rollno, name, marks in three subject). Write a program to create such an array and print out a list of students who have failed in more than one subject.

//Including all the required header files

#include<iostream>

using namespace std;

//Creating Class Student

class Student

{

    private:

    int rollNo[25],mark[25][3],nos,flag;

    string name[25];

    public:

    void inputData();

    void printFaildata(Student obj, int i);

    void failStudent(Student obj1,int j);

};

//Input all the student's data

void Student::inputData()

{

    Student student;

    int i, j;

    cout<<"Enter Number of Students : ";

    cin>>nos;

    for(i=0; i<nos; i++)

    {

        cout<<"Enter the details for Student No. "<<i+1<<" : \n";

        cout<<"Enter Roll Number : ";

        cin>>student.rollNo[i];

        cout<<"Enter Name : ";

        cin>>student.name[i];

        cout<<"Enter Marks of Three Subjects : ";

        for(j=0; j<3; j++)

        {

            cin>>student.mark[i][j];

        }

    }

    failStudent(student, nos);  //checking for student if they got fail mark in more than 1 subject

}

//Function to display student's who failed in more than 1 subject

void Student::failStudent(Student k, int totN)

{

    int i, j, fsub;

    k.flag=0;

    for(i=0; i<totN; i++ )

    {

        fsub=0;

        for(j=0; j<3; j++)

        {

            if (k.mark[i][j]<33)

            {

                fsub += 1;

            }

        }

        if(fsub>1)

        {

            k.flag=1;

            printFaildata( k, i); //Function for printing the data of student's failed in more than 1 subject.

        }

    }

    if(k.flag==0)   //if no student failed in more than 1 subject then prompt this message.

    {

        cout<<"\nNo Student Failed in more than 1 subject.\n\n";

    }

}

void Student::printFaildata( Student student, int n)

{

    int i;

    cout<<"\nStudent Roll Number is "<<student.rollNo[n];

    cout<<"\nStudent Name is "<<student.name[n];

    cout<<"\nStudent Marks are ";

    for(i=0; i<3; i++)

    {

        cout<<student.mark[n][i]<<"\t";

    }

}

int main()

{

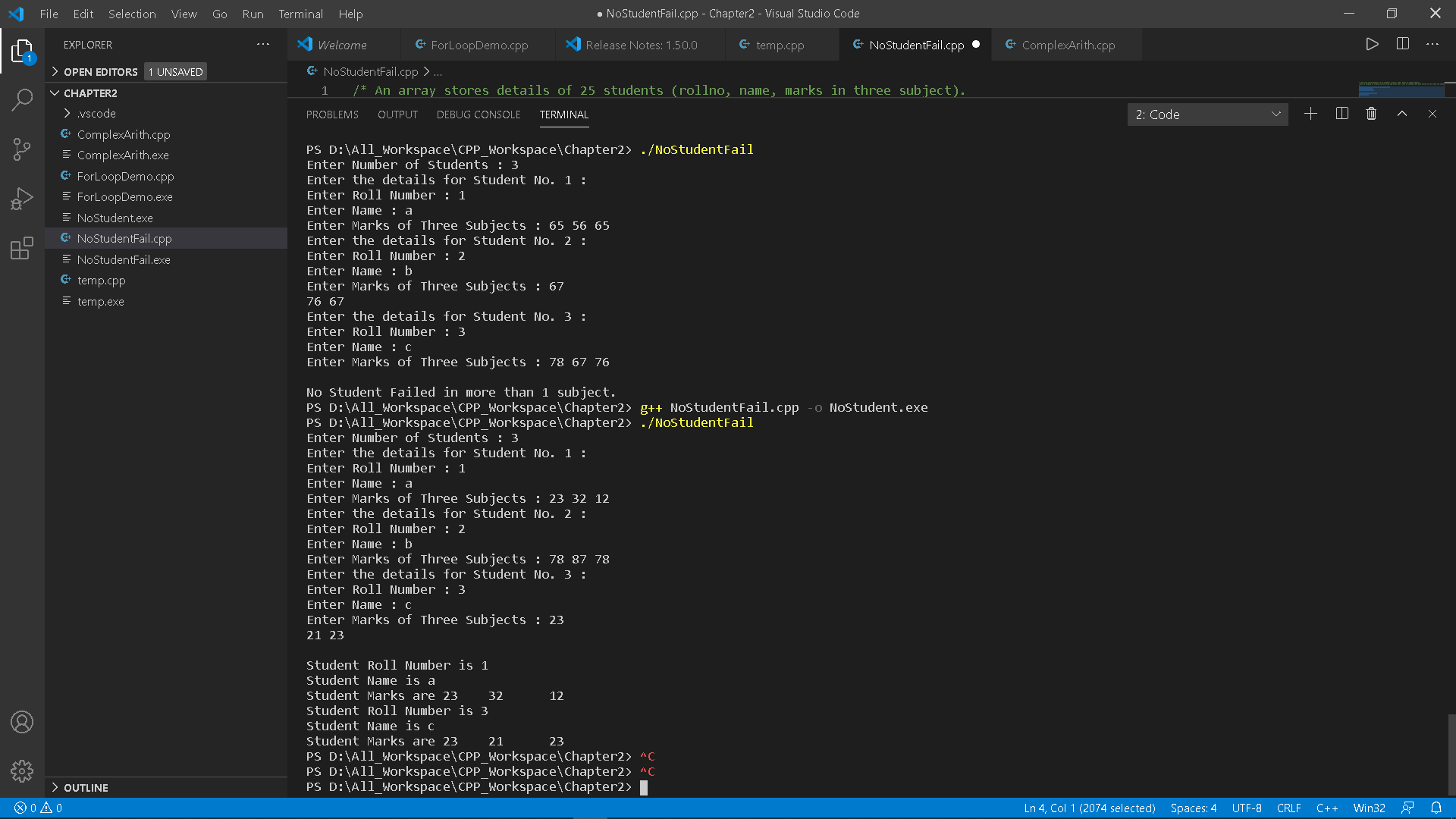
    Student stu;

    stu.inputData();

    return 0;

}

OUTPUT:



7. What should be output of below program? program is compiled on g++ compiler.

#include<iostream>

using namespace std;

struct student{

char a; char b; int c;

};

int main()

{

cout<<sizeof(student);

return 0;

}

Options:

(A) 4  
(B) 6  
(C) 8  
(D) 12

Give the Answer:

(C)

8. Which of the following statements assigns a value to the hourlyWage member of employee[2}?

Options:

(A) employee[2]->hourlyWage = 50.00;  
(B) employee2.hourlyWage = 7.50;  
(C) hourlyWage[2].employee = 29.75;  
(D) employee[2].hourlyWage = 75.00;

Give the answer:

(D)

9. Which of the following statements outputs the value of the gpa member of element 1 of the student array?

Options:

(A) cout<<student1.gpa;  
(B) cout<<firstStudent.gpa;  
(C) cout<<student[1].gpa;  
(D) cout<<student1 ->gpa;

Give the answer:

(C)

10. Which of the following statements outputs the value of the gpa member of element 1 of the student array?

Options:

(A) cout<<student1.gpa;  
(B) cout<<firstStudent.gpa;  
(C) cout<<student[1].gpa;  
(D) cout<<student1 ->gpa;

Give the answer:

(C)