**OOP MIDTERM PROGRAMMING QUESTIONS**

**Question2**

Describe Method Overloading? B. A point on the two-dimensional plane can be represented by two numbers: an x coordinate and a y coordinate. For example, (4,5) represents a point 4 units to the right of the vertical axis, and 5 units up from the horizontal axis. The sum of two points can be defined as a new point whose x coordinate is the sum of the x coordinates of the two points, and whose y coordinate is the sum of the y coordinates. Write a program that uses a structure called point to model a point. Define three points, and have the user input values to two of them. Then set the third point equal to the sum of the other two, and display the value of the new point. Interaction with the program might look like this: Enter coordinates for p1: 3 4 Enter coordinates for p2: 5 7 Coordinates of p1+p2 are: 8, 11

Overloadingallows different methods in a class to have the same name, but different signatures where the signature can differ by the number of input parameters or type of input parameters or both. Example in a class we can overload constructors containing different parameters like in class student if we have attributes like name, id , gender, grade so in one constructor we can initialize all the attributes and next we can also define the constructor where we initialize two constructors name and id.

struct point

{

int x;

int y;

};

int main()

{

point p1;

point p2;

point sum;

cout << "NIMRAH ALTAF ADAM SE-077" << endl;

cout << "Enter coordinates for p1: ";

cin >> p1.x;

cin >> p1.y;

cout << "Enter coordinates for p2: ";

cin >> p2.x;

cin >> p2.y;

sum.x = p1.x + p2.x;

sum.y = p1.y + p2.y;

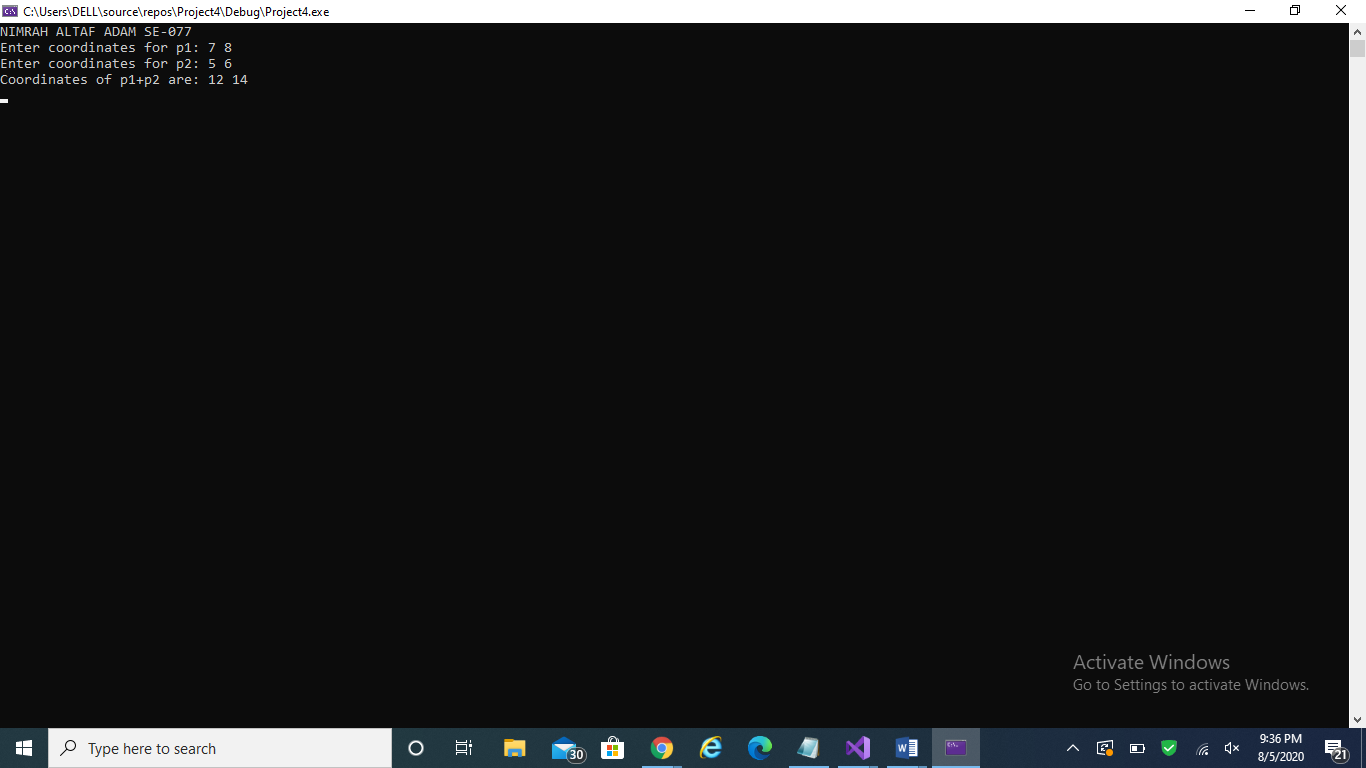
cout <<"Coordinates of p1+p2 are: " << sum.x << " " << sum.y << endl;

\_getch();

return(0);

}

OUTPUT:



Question 3c

 C. Write down a Employee Class having data members(Empno,Name,Address,Salary,Experience)with all the getter and setter methods for each data member and add one member function that can calculate the yearly increment of an employee depending on his/her grade,and one data member to display all the information of an employee

#include<iostream>

#include<conio.h>

#include<string>

using namespace std;

class Employee

{

private:

int Empno;

string Name;

string Address;

int Salary;

string Experience;

int Grade;

public:

int getEmpno()

{

return Empno;

}

void setEmpno(int e)

{

Empno = e;

}

string getName()

{

return Name;

}

void setName(string n)

{

Name = n;

}

string getAddress()

{

return Address;

}

void setAddress(string a)

{

Address = a;

}

int getSalary()

{

return Salary;

}

void setSalary(int s)

{

Salary = s;

}

string getExperience()

{

return Experience;

}

void setExperience(string ex)

{

Experience = ex;

}

int getGrade()

{

return Grade;

}

void setGrade(int g)

{

Grade = g;

}

/\*BPS - 16 to 22 Gazetted Officers / Commissioned Officers in Army / Navy / PAF and Junior Commissioned Officers of Armed Forces(JCOs) ...

BPS - 11 to 15 Non - Gazetted Officers / Junior Officers and Field Supervisors.

BPS - 5 to 10 Lower grade officials / Field work supervisors, SNCOs / NCOs.\*/

double increment(int s, int g)

{

if (g >= 5 && g <= 10)

{

return s + (s \* 0.05);

}

if (g >= 11 && g <= 15)

{

return s + (s \* 0.1);

}

if (g >= 16 && g <= 22)

{

return s + (s \* 0.2);

}

}

void ShowData()

{

cout << "The employee number is: " << Empno << endl;

cout << "The employee name is: " << Name << endl;

cout << "The employee address is: " << Address << endl;

cout << "The employee salary is: " << Salary << endl;

cout << "The employee experience is: " << Experience << endl;

cout << "The employee grade is: " << Grade << endl;

}

};

int main()

{

cout << "Nimrah Alatf Adam SE-077" << endl;

Employee e1;

e1.setEmpno(15);

e1.setName("Ahmed");

e1.setAddress("Gulshan House no: A/21 Block 7");

e1.setSalary(25000);

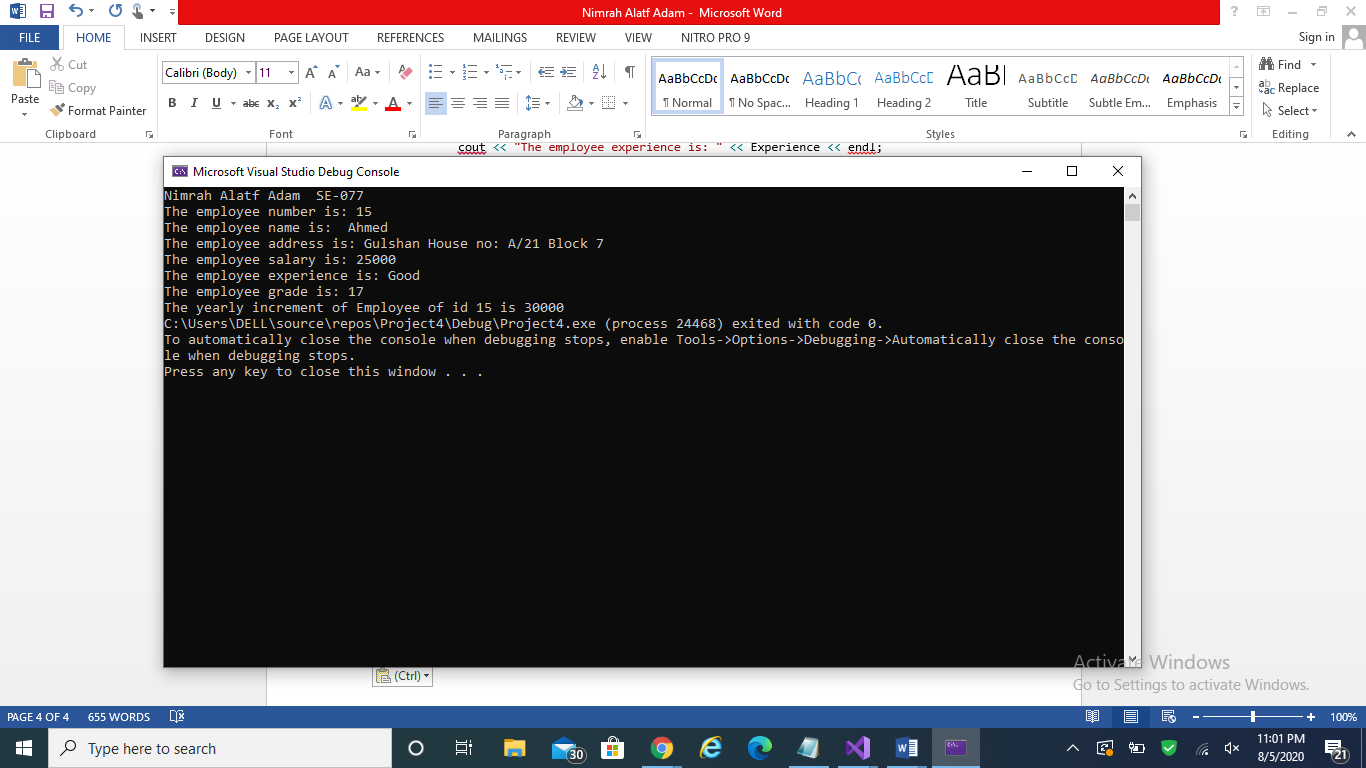
e1.setExperience("Good");

e1.setGrade(17);

e1.ShowData();

cout << "The yearly increment of Employee of id " << e1.getEmpno() << " is " << e1.increment(e1.getSalary(), e1.getGrade());

}



Question 4

Create a class called time that has separate int member data for hours, minutes, and seconds. One constructor should initialize this data to 0, and another should initialize it to fixed values. Another member function should display it, in 11:59:59 format. The final member function should add two objects of type time passed as arguments. A main() program should create two initialized time objects (should they be const?) and one that isn’t initialized. Then it should add the two initialized values together, leaving the result in the third time variable. Finally it should display the value of this third variable. Show\_data member function should display the time in hours, minutes seconds Make appropriate member functions const.

class time {

private:

int hours, minutes, seconds;

public:

time() {

hours = minutes = seconds = 0;

}

time(int h, int m, int s) {

hours = h;

minutes = m;

seconds = s;

}

void showTime() const {

cout << hours << ':' << minutes << ':' << seconds;

}

void addTime(time x, time y) {

seconds = x.seconds + y.seconds;

if (seconds > 59) {

seconds -= 60;

minutes++;

}

minutes += x.minutes + y.minutes;

if (minutes > 59) {

minutes -= 60;

hours++;

}

hours += x.hours + y.hours;

}

};

int main() {

const time a(2, 23, 45), b(4, 25, 15);

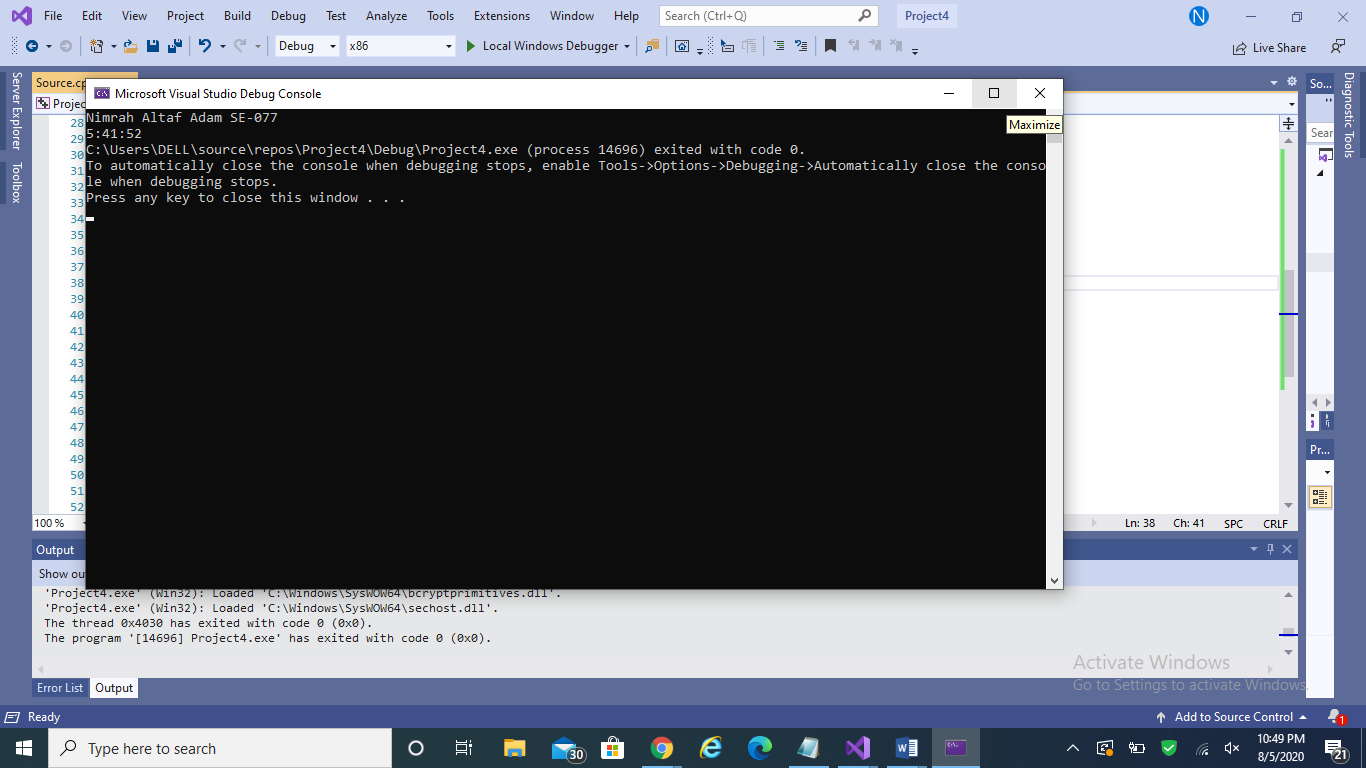
time c;

c.addTime(a, b);

c.showTime();

}

OUTPUT:



Q 5 Imagine a tollbooth at a bridge. Cars passing by the booth are expected to pay a 50 cent toll. Mostly they do, but sometimes a car goes by without paying. The tollbooth keeps track of the number of cars that have gone by, and of the total amount of money collected. Model this tollbooth with a class called tollBooth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both of these to 0. A member function called payingCar() increments the car total and adds 0.50 to the cash total. Another function, called nopayCar(), increments the car total but adds nothing to the cash total. Finally, a member function called display() displays the two totals. Make appropriate member functions const. Include a main program to test this class. This program should allow the user to count a paying car, and to count a nonpaying car. One member function should cause the program to print out the total cars and total cash .

class Tollbooth

{

private:

int TC = 0;

double M = 0.0;

public:

void defalut()

{

TC = 0;

M = 0.0;

};

void paycar()

{

M += 0.5;

TC += 1;

};

void notpaycar()

{

TC += 1;

};

double display()

{

cout << "total cars " << TC << endl;

cout << "total cash " << M << endl;

return 0.0;

};

};

int main()

{

cout << "Nimrah Altaf Adam SE-077" << endl;

Tollbooth car1;

car1.paycar();

car1.notpaycar();

car1.display();

return 0;

}

OUTPUT:

