ASSIGNMENT PRACTICAL - 1

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(1):Demonstrate the use of static variables in a class by using it to count the number of objects created in the program.

****
****

```
#include<iostream>
using namespace std;

class statt{
  static int a;
  public:
  statt()
  {
    ++a;
  }
  void print(void)
  {
    cout << "Count : " << a << endl;
```

```
   }
};

int statt::a;
int main()
{
   statt s,s1;
   s.print();
   statt s2;
   s.print();
   return 0;
}
```

****
****

output:

```
Count : 2
Count : 3
```

****
****

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(2):Define a class to represent a bank account. Include the following members :

DATA MEMBERS	MEMBER FUNCTIONS
Name of depositor	(1) To assign initial values
Account Number	(2) To Deposit the amount
Type of Account	(3) To withdraw an amount after checking the Balance amount in account
	(4) To display name and balance

Write C++ program to handle 10 customers.

****
****

```
#include<iostream>
#include<string>
using namespace std;

class bank_account
{
   string Name_of_Depositor;
```

```
int Account_Number;
string Account_Type;
int Balance;
public:
void initial_values()
{
   cout << endl << "Enter Name Of Depositor : ";
   cin >> Name_of_Depositor;
   cout << "Enter The Account Number : ";
   cin >> Account_Number;
   cout << "Enter The Account Type : ";
   cin >> Account_Type;
   cout << "Enter The Balance : ";
   cin >> Balance;
}
void Deposit()
{
   int amount;
   cout << "Enter The Amount You Want To Add : ";
   cin >> amount;
   Balance += amount;
}

void Withdraw()
{
   int amount;
   cout << "Enter The Amount You Want To Withdraw : ";
   cin >> amount;
   if( Balance < amount )
   {
      cout << "Insufficient Balance" << endl;
```

```
        }
        else{
            Balance -= amount;
        }
    }

    void Display()
    {
        cout << "Name : " << Name_of_Depositor << endl << "Balance : " << Balance << endl;
    }

    int check(int acc,int i)
    {
        if( Account_Number == acc )
        {
            return i;
        }
        else{
            return -1;
        }
    }
};

void menu(class bank_account []);

int main()
{
    int n,index;
    bank_account s[10];
    for (int i = 0; i < 10; i++)
    {
        cout << "Enter Detail For "<< i+1 << " Customer" << endl;
        s[i].initial_values();
    }
    menu(s);
    return 0;
}

void menu(class bank_account s[])
{
    int n,account_no,index = -1;
    cout << endl << " 1 . Deposit Amount." << endl << " 2 . Withdraw Amount." << endl << " 3 . Show
Balnace." << endl << " 4 . Exit" << endl;
    cin >> n;
    switch (n)
    {
    case(1):
        cout << "Enter Your Account_No : ";
        cin >> account_no;
        for( int i = 0; i < 10; i++ )
        {
            index = s[i].check(account_no,i);
            if( index != -1 )
            {
                i = 10;
            }
        }
        if(index == -1)
        {
            cout << "InValid Account Number" << endl;
        }
        else{
```

```
        s[index].Deposit();
    }
    menu(s);
case(2):
    cout << "Enter Your Account_No : ";
    cin >> account_no;
    for( int i = 0; i < 10; i++ )
    {
        index = s[i].check(account_no,i);
        if( index != -1 )
        {
            i = 10;
        }
    }
    if(index == -1)
    {
        cout << "InValid Account Number" << endl;
    }
    else{
        s[index].Withdraw();
    }
    menu(s);
case(3):
    cout << "Enter Your Account_No : ";
    cin >> account_no;
    for( int i = 0; i < 10; i++ )
    {
        index = s[i].check(account_no,i);
        if( index != -1 )
        {
            i = 10;
            }
        }
        if(index == -1)
        {
            cout << "InValid Account Number" << endl;
        }
        else{
            s[index].Display();
        }
        menu(s);
    case(4):
        exit(0);
    default:
        cout << "Invalid Call" << endl;
        menu(s);
    }
}
```

****
****

Output:

```
Enter Detail For 1 Customer
Enter Name Of Depositor : Pradip
Enter The Account Number : 789
Enter The Account Type : Saving
Enter The Balance : 3400

Enter Detail For 2 Customer
Enter Name Of Depositor : Lakshya
Enter The Account Number : 678

Enter The Account Type : Saving

Enter The Balance : 2000


Enter Detail For 3 Customer

Enter Name Of Depositor : Ajinkya

Enter The Account Number : 123

Enter The Account Type : Saving

Enter The Balance : 4000


Enter Detail For 4 Customer

Enter Name Of Depositor : Nirav

Enter The Account Number : 456

Enter The Account Type : Current

Enter The Balance : 5000


Enter Detail For 5 Customer

Enter Name Of Depositor : Sudip

Enter The Account Number : 367

Enter The Account Type : Current

Enter The Balance : 8700


Enter Detail For 6 Customer

Enter Name Of Depositor : Monil

Enter The Account Number : 269

Enter The Account Type : Current

Enter The Balance : 4500


Enter Detail For 7 Customer

Enter Name Of Depositor : Tejas

Enter The Account Number : 120
Enter The Account Type : Current
Enter The Balance : 10000

Enter Detail For 8 Customer
Enter Name Of Depositor : Dhaval
Enter The Account Number : 638
Enter The Account Type : Saving
Enter The Balance : 900

Enter Detail For 9 Customer
Enter Name Of Depositor : Aditya
Enter The Account Number : 333
Enter The Account Type : Saving
Enter The Balance : 9000

Enter Detail For 10 Customer
Enter Name Of Depositor : Ravi
Enter The Account Number : 459
Enter The Account Type : Current
Enter The Balance : 16000

 1 . Deposit Amount.
 2 . Withdraw Amount.
 3 . Show Balnace.
 4 . Exit
1
Enter Your Account_No : 333
Enter The Amount You Want To Add : 500

 1 . Deposit Amount.
 2 . Withdraw Amount.
 3 . Show Balnace.
 4 . Exit
3
Enter Your Account_No : 333
Name : Aditya
Balance : 9500

 1 . Deposit Amount.
 2 . Withdraw Amount.
 3 . Show Balnace.
 4 . Exit
2
Enter Your Account_No : 333
Enter The Amount You Want To Withdraw : 10000
Insufficient Balance

 1 . Deposit Amount.
 2 . Withdraw Amount.
 3 . Show Balnace.
 4 . Exit
2
Enter Your Account_No : 333
Enter The Amount You Want To Withdraw : 450

 1 . Deposit Amount.
 2 . Withdraw Amount.
 3 . Show Balnace.
 4 . Exit
3
Enter Your Account_No : 333
Name : Aditya
```

```
Balance : 9050

 1 . Deposit Amount.
 2 . Withdraw Amount.
 3 . Show Balnace.
 4 . Exit
4
```

****
****

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(3):Create class STUDENT having rollno, name and age as data members, also take subject with three subjects and initialize their value with minimum

passing marks. Using member function, modify marks of student with specific rollno which is given by user.

****
****

```
#include<iostream>

#include<string>
```

```
using namespace std;

class student{
        int rno;
        string name;
        int age;
        float marks[3] = {35,35,35};

        public:
        void addstd()
        {
                cout << endl << "Enter Student Roll No : " << endl;
                cin >> rno;
                cout << endl << "Enter Student Name : " << endl;
                cin >> name;
                cout << endl << "Enter Student Age : " << endl;
                cin >> age;
        }

        int check(int roll,int i)
        {
                if(roll == rno)
                {
                        return i;
                }
                else
                {
                        return -1;
                }
        }
        void edit()
        {
                cout << endl << "Enter Marks For Subject 1  :" << endl;
                cin     >> marks[0];
                cout << endl << "Enter Marks For Subject 2  :" << endl;
                cin     >> marks[1];
                cout << endl << "Enter Marks For Subject 3  :" << endl;
                cin     >> marks[2];
        }

        void display()
        {
                cout << endl << "Name : " << name << endl;
                cout << endl << "Roll No : " << rno << endl;
                cout << endl << "Age : " << age <<endl;
                cout << endl << "Subject 1 : " << marks[0] << endl;
                cout << endl << "Subject 2 : " << marks[1] << endl;
                cout << endl << "Subject 3 : " << marks[2] << endl;
        }
};

void menu(class student []);

int main()
{
        student std[10];
        menu(std);
        return 0;
}

void menu(class student std[])
```

```
{
    int index = 0,n,i,checkroll = 0,editindex = -1;
    do{
        cout << " 1 . Add Student." << endl << " 2 . Edit Student Details." << endl << " 3 . Show Details of Student." << endl << " 4 . Exit" << endl;
    cin >> n;

        if( n == 1){
            std[index].addstd();
            cout << endl << "Student Added Sucessfully With Default Marks." << endl;
        }
        else if( n == 2)
        {
            cout << endl << "Enter Student Roll No" << endl;
            cin >> checkroll;
            for( i = 0; i <= index; i++ )
            {
                editindex = std[i].check(checkroll,i);
                if( editindex != -1 )
                {
                    i = index+1;
                }
            }
            if( editindex == -1 )
            {
                cout << endl << "Invalid Roll No." << endl;
            }
            else{
                std[editindex].edit();
                cout << endl << "Student Marks Updated Sucessfully." << endl;
            }
        }
        else if( n == 3 )
        {
            cout << endl << "Enter Student Roll No" << endl;
            cin >> checkroll;
            for( i = 0; i <= index; i++ )
            {
                editindex = std[i].check(checkroll,i);
                if( editindex != -1 )
                {
                    i = index+1;
                }
            }
            if( editindex == -1 )
            {
                cout << endl << "Invalid Roll No." << endl;
            }
            else{
                std[editindex].display();
            }
        }
        else{
            exit(0);
        }
    }while(n < 4);
}
```

****
****

output:

```
 1 . Add Student.
 2 . Edit Student Details.
 3 . Show Details of Student.
 4 . Exit
1

Enter Student Roll No :
10

Enter Student Name :
pradip

Enter Student Age :
22

Student Added Sucessfully With Default Marks.

 1 . Add Student.
 2 . Edit Student Details.
 3 . Show Details of Student.
 4 . Exit
1

Enter Student Roll No :
7

Enter Student Name :
lakshya

Enter Student Age :
```

```
21

Student Added Sucessfully With Default Marks.

 1 . Add Student.
 2 . Edit Student Details.
 3 . Show Details of Student.
 4 . Exit
3

Enter Student Roll No
7

Name : lakshya

Roll No : 7

Age : 21

Subject 1 : 35

Subject 2 : 35

Subject 3 : 35
 1 . Add Student.
 2 . Edit Student Details.
 3 . Show Details of Student.
 4 . Exit
2

Enter Student Roll No
```

```
7

Enter Marks For Subject 1  :
39

Enter Marks For Subject 2  :
41

Enter Marks For Subject 3  :
57

Student Marks Updated Sucessfully.
 1 . Add Student.
 2 . Edit Student Details.
 3 . Show Details of Student.
 4 . Exit
3

Enter Student Roll No
7

Name : lakshya

Roll No : 7

Age : 21



Subject 3 : 57
```

****
****

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(4):Create a class Rectangle. The class has attributes length and width, each of which defaults to 1. It has member functions that calculate the perimeter and the

area of the rectangle. It has set and get functions for both length and width. The set functions should verify that length and width are each floating-point

numbers larger than 0.0 and less than 20.0.

****
****

```
#include<iostream>

using namespace std;


class rectangle

{

        float len,wid,perimeter,area;


public:

        rectangle()
```

```
{
        len = 1;
        wid = 1;
}

void set()
{
        cout << "Enter length of Rectangle : ";
        cin >> len;
        while( len < 0 || len > 20 )
        {
                cout << "Length Must between 0.0 and 20.0" << endl;
                cin >> len;
        }

        cout << "Enter Width of Rectangle : ";
        cin >> wid;
        while( wid < 0 || wid > 20)
        {
                cout << "Width Must between 0.0 and 20.0" << endl;
                cin >> wid;
        }
}

void cal_area()
{
        area = len + wid;
}

void cal_peri()
{
```

```
                perimeter = 2 * ( len + wid );
        }

        void get()
        {
                cout << endl << "Area of Ractangle : " << area << endl;
                cout << endl << "Perimeter of Rectangle : " << perimeter << endl;
        }

};

void main()
{
        rectangle rec;
        rec.set();
        rec.cal_area();
        rec.cal_peri();
        rec.get();
}
```

****
****

output:

```
Enter length of Rectangle : 25
Length Must between 0.0 and 20.0
7
Enter Width of Rectangle: 100
Width Must between 0.0 and 20.0
12
```

```
Area of Ractangle : 19

Perimeter of Rectangle : 38
```

****
****

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(5):Define a supplier class. The class contains details about the suppliers. One of the details is the list of items supplied by

the supplier. Create a class Item to store item details. The items supplied by any given supplier are different and varying.

Use dynamic memory allocation in the constructor function to achieve the solution.

****
****

```
#include<iostream>

#include<string>


using namespace std;
```

```
class item

{

        string it_name;

        int it_id;

public:

        void read()

        {

                cout<<"Enter item id: ";

                cin>>it_id;

                cout<<"Enter Item Name: ";

                cin>>it_name;

        }

        void display()

        {

                cout<<"\nItem id= "<<it_id<<endl;

                cout<<"Item Name= "<<it_name<<endl;

        }


};


class supplier

{

        string sp_name;

        int sp_id,n;

        item *it;

public:

        void read()

        {

                cout << "Enter Supplier id : ";

                cin >> sp_id;
                cout << "Enter Supplier Name : ";
                cin >> sp_name;
                cout << "Enter Number of items supplier supply: ";
                cin >> n;
                it = new item[n];
                for( int i = 0; i < n; i++)
                {
                    it[i].read();
                }
        }
        void display()
        {
                cout << "Supplier ID : " << sp_id << endl;
                cout << "Supplier Name : " << sp_name << endl;
                cout << endl << "Items Supplied by supplier" << endl;
                for( int i = 0; i < n; i++)
                {
                    it[i].display();
                }
        }
};

void main()
{
        supplier s[2];
        s[0].read();
        s[1].read();
        s[0].display();
        s[1].display();
}
```

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

output:

```
Enter Supplier id : 1

Enter Supplier Name : Pradip

Enter Number of items supplier supply : 1


Enter item id : 1

Enter Item Name : Games


Enter Supplier id : 2

Enter Supplier Name : Sudip

Enter Number of items supplier supply : 2


Enter item id : 2

Enter Item Name : Laptop


Enter item id : 3

Enter Item Name : Speakers


Supplier ID : 1

Supplier Name : Pradip


Items Supplied by supplier

Item id = 1

Item Name = Gmaes


Supplier ID: 2

Supplier Name: Sudip
```

```
Items supplied by supplier

Item id = 2

Item Name = Laptop


Item id = 3

Item Name = Speakers
```

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(6):Define a class Student. Add data members as Rollno, Name, Marks_obtained, Max_marks and Percentage. Write member functions for

reading values, calculating percentage and printing values of student. Define one more class as MCA_II. MCA_II contains array

of students. MCA_II class should contain member functions as Add, delete, modify and replace.

MCA_II is to be defined as friend of Student class.

****
****

```
#include<iostream>

#include<string>
```

```
using namespace std;

class student
{
	int roll_no;
	float marks_obtained,max_marks,percentage;
	string name;

public:
	void read()
	{
		cout<<"\n==============================================\n";
		cout<<"Enter Roll Number: ";
		cin>>roll_no;
		cout<<"Enter Name: ";
		cin>>name;
		cout<<"Enter Maximum marks: ";
		cin>>max_marks;
		cout<<"Enter marks obtained: ";
		cin>>marks_obtained;
	}

	float calc_per()
	{
		percentage=(100*marks_obtained)/max_marks;
		return percentage;
	}

	void display()
	{
		cout<<"\nRoll number: "<<roll_no<<endl;
```

```
                cout<<"Name: "<<name<<endl;
                cout<<"Marks obtained out of "<<max_marks<<" is "<<marks_obtained<<endl;
                cout<<"Percentage: "<<calc_per()<<"%"<<endl;
        }
        friend class mca2;
};

class mca2
{
        student s[10];
        static int limit;
public:
        void add()
        {
                int count,i;
                cout<<"Enter How many students you want to add: ";
                cin>>count;
                for(i=limit;i<limit+count;i++)
                {
                        s[i].read();
                }
                limit+=count;
        }
        void Delete()
        {
                int i,r_no;
                bool flag=false;
                cout<<"Enter roll number of which student you want to delete: ";
                cin>>r_no;
                for(i=0;i<limit;i++)
                {
```

```
		if(s[i].roll_no==r_no)
		{
			flag=true;
			while(i<limit-1)
			{
				s[i]=s[i+1];
				i++;
			}
			limit--;
			break;
		}
	}
	if(!flag)
	{
		cout<<"\n-------------------Invalid roll number!!\n";
	}
}
void modify()
{
	int i,r_no;
	bool flag=false;
	cout<<"Enter Roll number of student which you want to modify: "<<endl;
	cin>>r_no;
	for(i=0;i<limit;i++)
	{
		if(s[i].roll_no==r_no)
		{
			flag=true;
			cout<<"Enter maximum marks: ";
			cin>>s[i].max_marks;
			cout<<"Enter marks obtained: ";
```

```
                cin>>s[i].marks_obtained;
                break;
            }
        }

        if(!flag)
        {
            cout<<"\n-----------------Invalid roll number!!\n";
        }
    }
    void replace()
    {
        int i,j,r_no;
        bool flag=false;
        cout<<"Enter Roll number of student which you want to replace: "<<endl;
        cin>>r_no;

        for(i=0;i<limit;i++)
        {
            if(s[i].roll_no==r_no)
            {
                flag=true;
                cout<<"Enter roll number of student which you want to replace with: ";
                cin>>r_no;
                for(j=0;j<limit;j++)
                {
                    if(s[j].roll_no==r_no)
                    {
                        s[i]=s[j];
                        break;
                    }
                }
                while(j<limit-1)
                {
                    s[j]=s[j+1];
                    j++;
                }
                limit--;
            }
        }
        if(!flag)
        {
            cout<<"\n------------------Invalid roll number!!\n";
        }
    }
    void show()
    {
        if(limit==0)
            cout<<"\n-------------List is Empty!!!";
        else
        {
            for(int i=0;i<limit;i++)
            {
                s[i].display();
            }
        }
    }
};

int mca2::limit=0;
```

```
 void main()
{
        int key;
        mca2 m;
        do
        {
                cout<<"--------------------------------------------------\n";
                cout<<"1-Add Students\n2-Delete Students\n3-Modify Marks of student";
                cout<<"\n4-Replace student\n5-Display Students\n6-Exit\nEnter Your Choice: ";
                cin>>key;
                if(key==1)
                        m.add();
                else if(key==2)
                        m.Delete();
                else if(key==3)
                        m.modify();
                else if(key==4)
                        m.replace();
                else if(key==5)
                        m.show();
        }while(key!=6);
}
```

****
****

output:

```
--------------------------------------------------
1-Add Students
2-Delete Students
3-Modify Marks of student
4-Replace student
```

```
5-Display Students

6-Exit

Enter Your Choice: 1

Enter How many students you want to add: 2


================================================

Enter Roll Number: 10

Enter Name: Pradip

Enter Maximum marks: 500

Enter marks obtained: 495


================================================

Enter Roll Number: 05

Enter Name: Nirav

Enter Maximum marks: 500

Enter marks obtained: 499

--------------------------------------------------------

1-Add Students

2-Delete Students

3-Modify Marks of student

4-Replace student

5-Display Students

6-Exit

Enter Your Choice: 5


Roll number: 10

Name: Pradip

Marks obtained out of 500 is 495

Percentage: 99.8%


Roll number: 5
```

```
Name: Nirav

Marks obtained out of 500 is 499

Percentage: 98%

--------------------------------------------------

1-Add Students

2-Delete Students

3-Modify Marks of student

4-Replace student

5-Display Students

6-Exit

Enter Your Choice: 1

Enter How many students you want to add: 1


=================================================

Enter Roll Number: 07

Enter Name: Lakshya

Enter Maximum marks: 500

Enter marks obtained: 480

--------------------------------------------------

1-Add Students

2-Delete Students

3-Modify Marks of student

4-Replace student

5-Display Students

6-Exit

Enter Your Choice: 5


Roll number: 10

Name: Pradip

Marks obtained out of 500 is 495

Percentage: 99.8%
```

```
Roll number: 5
Name: Nirav
Marks obtained out of 500 is 499
Percentage: 98%

Roll number: 7
Name: Lakshya
Marks obtained out of 500 is 480
Percentage: 96%
----------------------------------------------------
1-Add Students
2-Delete Students
3-Modify Marks of student
4-Replace student
5-Display Students
6-Exit

Enter Your Choice: 2
Enter roll number of which student you want to delete: 50
Invalid roll number!!


----------------------------------------------------
1-Add Students
2-Delete Students
3-Modify Marks of student
4-Replace student
5-Display Students
6-Exit
Enter Your Choice: 2
Enter roll number of which student you want to delete: 10
```

```
--------------------------------------------------
1-Add Students

2-Delete Students

3-Modify Marks of student

4-Replace student

5-Display Students

6-Exit

Enter Your Choice: 5


Roll number: 5

Name: Nirav

Marks obtained out of 500 is 499

Percentage: 98%


Roll number: 7

Name: Lakshya

Marks obtained out of 500 is 480

Percentage: 96%
--------------------------------------------------
1-Add Students

2-Delete Students

3-Modify Marks of student

4-Replace student

5-Display Students

6-Exit

Enter Your Choice: 3

Enter Roll number of student which you want to modify:

7

Enter maximum marks: 500

Enter marks obtained: 485
--------------------------------------------------
```

```
1-Add Students

2-Delete Students

3-Modify Marks of student

4-Replace student

5-Display Students

6-Exit

Enter Your Choice: 5


Roll number: 5

Name: Nirav

Marks obtained out of 500 is 499

Percentage: 98%


Roll number: 7

Name: Lakshya

Marks obtained out of 500 is 485

Percentage: 97%

----------------------------------------------------

1-Add Students

2-Delete Students

3-Modify Marks of student

4-Replace student

5-Display Students

6-Exit

Enter Your Choice: 6
```

****
****

****
****

Name : Pradip S Karmakar.

Roll No : 10

Class : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(7):Define a class Car. Add data members as Make, Color, Size, and Cost. Write member functions for reading values and

printing values of car. Define one more class as CarCollection. CarCollection contains array of cars. CarCollection class

should contain member functions as Add, delete, modify and replace. CarCollection is to be defined as friend of Car class.

Write C++ programs to test your classes.

****
****

```
#include<iostream>

#include<conio.h>

#include<string>


using namespace std;


class Car

{

            string maker,colour;

            float size,cost;

            int id;


public:
		void read()
		{
			cout<<"\n---------------------\n";
			cout<<"Enter id of car: ";
			cin>>id;
			cout<<"Enter Maker of car: ";
			cin>>maker;
			cout<<"Enter colour of car: ";
			cin>>colour;
			cout<<"Enter size of car in inches: ";
			cin>>size;
			cout<<"Enter cost of car in Rupee: ";
			cin>>cost;
		}

		void display()
		{
			cout<<"\n======================\n";
			cout<<"Car ID: "<<id<<endl;
			cout<<"Maker of car: "<<maker<<endl;
			cout<<"Colour of car: "<<colour<<endl;
			cout<<"Size of car: "<<size<<" Inches"<<endl;
			cout<<"Cost of car: "<<cost<<" Rupee"<<endl;
		}
		friend class CarCollection;
};

class CarCollection
{
		Car c[10];
		static int limit;
```

```
public:

    void add()
    {
        int count;
        cout<<"Enter HOw many Cars you want to add: ";
        cin>>count;
        for(int i=limit;i<limit+count;i++)
        {
            c[i].read();
        }
        limit+=count;
    }

    void remove()
    {
        int c_id,flag=0,j;
        cout<<"Enter Car id which You want to delete: ";
        cin>>c_id;
        for(int i=0;i<limit && !flag;i++)
        {
            if(c_id==c[i].id)
            {
                while(i<limit-1)
                {
                    c[i]=c[i+1];
                    i++;
                }
                limit--;
                flag=1;
            }
```

```
          }


          if(!flag)
                    cout<<"\n----------------------Invalid Car ID!!!!!";
}


void modify()
{
          int c_id,flag=0;
          cout<<"Enter Car id which You want to modify: ";
          cin>>c_id;
          for(int i=0;i<limit && !flag;i++)
          {
                    if(c_id==c[i].id)
                    {
                              cout<<"Enter Maker name: ";
                              cin>>c[i].maker;
                              cout<<"Enter colour of car: ";
                              cin>>c[i].colour;
                              cout<<"Enter size of car in inches: ";
                              cin>>c[i].size;
                              cout<<"Enter cost of car in Rupee: ";
                              cin>>c[i].cost;
                              flag=1;
                    }
          }


          if(!flag)
                    cout<<"\n----------------------Invalid Car ID!!!!!";
}
```

```
void replace()
{
    int c_id1,c_id2,j,flag=0;
    cout<<"Enter Car id which You want to replace: ";
    cin>>c_id1;
    cout<<"Enter Car id which You want to replace with: ";
    cin>>c_id2;
    for(int i=0;i<limit && !flag;i++)
    {
        if(c_id1==c[i].id)
        {
            for(j=0;j<limit;j++)
            {
                if(c_id2==c[j].id)
                {
                    flag=1;
                    c[i]=c[j];
                    while(j<limit-1)
                    {
                        c[j]=c[j+1];
                        j++;
                    }
                    limit--;
                }
            }


        }
    }

    if(!flag)
```

```
                        cout<<"\nInvalid Car ID";
        }
        void display()
        {
                for(int i=0;i<limit;i++)
                        c[i].display();
        }

};

int CarCollection::limit=0;
void main()
{
        CarCollection c;
        int key;
        do
        {
                cout<<"---------------------------------------------------------\n";
                cout<<"1-Add Car\n2-Delete Car\n3-Modify Car";
                cout<<"\n4-Replace Car\n5-Display Car\n6-Exit\nEnter Your Choice: ";
                cin>>key;
                if(key==1)
                        c.add();
                else if(key==2)
                        c.remove();
                else if(key==3)
                        c.modify();
                else if(key==4)
                        c.replace();
                else if(key==5)
                        c.display();
```

```
	}while(key!=6);


	getch();
}
```

```
*******************************************************************************************
*************************************************************
output:
--------------------------------------------------
1-Add Car
2-Delete Car
3-Modify Car
4-Replace Car
5-Display Car
6-Exit
Enter Your Choice: 1
Enter HOw many Cars you want to add: 2

---------------------
Enter id of car: 1
Enter Maker of car: Maruti
Enter colour of car: red
Enter size of car in inches: 23
Enter cost of car in Rupee: 150000

---------------------
Enter id of car: 2
Enter Maker of car: Suzuki
Enter colour of car: white
Enter size of car in inches: 20
Enter cost of car in Rupee: 200000
```

```
---------------------------------------------------
1-Add Car
2-Delete Car
3-Modify Car
4-Replace Car
5-Display Car
6-Exit
Enter Your Choice: 5

======================
Car ID: 1
Maker of car: Maruti
Colour of car: red
Size of car: 23 Inches
Cost of car: 150000 Rupee

======================
Car ID: 2
Maker of car: Suzuki
Colour of car: white
Size of car: 20 Inches
Cost of car: 200000 Rupee
---------------------------------------------------
1-Add Car
2-Delete Car
3-Modify Car
4-Replace Car
5-Display Car
6-Exit
Enter Your Choice: 1
Enter HOw many Cars you want to add: 2
```

```
----------------------
Enter id of car: 3
Enter Maker of car: Mahindra
Enter colour of car: blue
Enter size of car in inches: 30
Enter cost of car in Rupee: 3500000

----------------------
Enter id of car: 4
Enter Maker of car: ferrari
Enter colour of car: red
Enter size of car in inches: 24
Enter cost of car in Rupee: 4500000
----------------------------------------------------
1-Add Car
2-Delete Car
3-Modify Car
4-Replace Car
5-Display Car
6-Exit
Enter Your Choice: 5

======================
Car ID: 1
Maker of car: Maruti
Colour of car: red
Size of car: 23 Inches
Cost of car: 150000 Rupee

======================
Car ID: 2

Maker of car: Suzuki

Colour of car: white

Size of car: 20 Inches

Cost of car: 200000 Rupee


======================

Car ID: 3

Maker of car: Mahindra

Colour of car: blue

Size of car: 30 Inches

Cost of car: 3.5e+006 Rupee


======================

Car ID: 4

Maker of car: ferrari

Colour of car: red

Size of car: 24 Inches

Cost of car: 4.5e+006 Rupee

----------------------------------------------------

1-Add Car

2-Delete Car

3-Modify Car

4-Replace Car

5-Display Car

6-Exit

Enter Your Choice: 2

Enter Car id which You want to delete: 2

----------------------------------------------------

1-Add Car

2-Delete Car
```

```
3-Modify Car
4-Replace Car
5-Display Car
6-Exit
Enter Your Choice: 5

======================
Car ID: 1
Maker of car: Maruti
Colour of car: red
Size of car: 23 Inches
Cost of car: 150000 Rupee

======================
Car ID: 3
Maker of car: Mahindra
Colour of car: blue
Size of car: 30 Inches
Cost of car: 3.5e+006 Rupee

======================
Car ID: 4
Maker of car: ferrari
Colour of car: red
Size of car: 24 Inches
Cost of car: 4.5e+006 Rupee
--------------------------------------------------
1-Add Car
2-Delete Car
3-Modify Car
4-Replace Car
```

```
5-Display Car
6-Exit
Enter Your Choice: 3
Enter Car id which You want to modify: 1
Enter Maker name: maruti
Enter colour of car: red
Enter size of car in inches: 18
Enter cost of car in Rupee: 160000
---------------------------------------------------
1-Add Car
2-Delete Car
3-Modify Car
4-Replace Car
5-Display Car
6-Exit
Enter Your Choice: 4
Enter Car id which You want to replace: 6
Enter Car id which You want to replace with: 2
Invalid Car ID
---------------------------------------------------
1-Add Car
2-Delete Car
3-Modify Car
4-Replace Car
5-Display Car
6-Exit
Enter Your Choice: 4
Enter Car id which You want to replace: 3
Enter Car id which You want to replace with: 4
---------------------------------------------------
1-Add Car
```

```
2-Delete Car

3-Modify Car

4-Replace Car

5-Display Car

6-Exit

Enter Your Choice: 5


======================

Car ID: 1

Maker of car: maruti

Colour of car: red

Size of car: 18 Inches

Cost of car: 160000 Rupee


======================

Car ID: 4

Maker of car: ferrari

Colour of car: red

Size of car: 24 Inches

Cost of car: 4.5e+006 Rupee

---------------------------------------------------

1-Add Car

2-Delete Car

3-Modify Car

4-Replace Car

5-Display Car

6-Exit

Enter Your Choice:
```

****
****

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(8):Use Employee and EmpCollection classes. Employee class contains details about employee and EmpCollection contains

collection of employees in form of an array. Provide GetSubordinates friend function which returns an object of

EmpCollection class which contains details of subordinates of a manager. The employee object describing manager is to be

passed as a parameter.

****
****

```
#include<iostream>

#include<string>

#include<string.h>


using namespace std;


class Employee

{

        string name,dept;
        int id;
public:
        string designation;
        void input()
        {
                cout<<"\n-----------------------\n";
                cout<<"Enter Employee ID: ";
                cin>>id;
                cout<<"Enter Employee Name: ";
                cin>>name;
                cout<<"Enter Department of employee: ";
                cin>>dept;
                cout<<"Enter Designation Of Employee:";
                cin>>designation;
        }

        void output()
        {
                cout<<"\n-----------------\n";
                cout<<"Employee ID: "<<id<<endl;
                cout<<"Employee Name: "<<name<<endl;
                cout<<"Employee Department: "<<dept<<endl;
                cout<<"Employee Designation: "<<designation<<endl;


        }

};

class EmpCollection
{
        Employee e[10];
```

```
		int limit;
	public:
		void input()
		{
			cout<<"Enter number of Employees: ";
			cin>>limit;
			for(int i=0;i<limit;i++)
				e[i].input();
		}

		void output()
		{
			cout<<"\n===========================================\n";
			for(int i=0;i<limit;i++)
			{
				e[i].output();
			}
		}

		friend EmpCollection GetSubordinates(EmpCollection);
	};

	EmpCollection GetSubordinates(EmpCollection e1)
	{
		int i,j=0;
		EmpCollection e2;
		for(i=0;i<e1.limit;i++)
		{
			if(e1.e[i].designation !="Manager")
			{
				e2.e[j]=e1.e[i];
```

```
                        j++;
                }
        }
        e2.limit=j;
        return e2;
}
void main()
{
        EmpCollection e1,e3;
        e1.input();
        e1.output();
        e3=GetSubordinates(e1);
        cout<<"\n-------------------------Details of subordinates of Manager :"<<endl;
        e3.output();
}
```

****
****

****
****

```
output:
Enter number of Employees: 4

----------------------
Enter Employee ID: 10
Enter Employee Name: Pradip
Enter Department of employee: IT
Enter Designation Of Employee:senior Developer

----------------------
Enter Employee ID: 7
Enter Employee Name: Lakshya
Enter Department of employee: IT
Enter Designation Of Employee:junior developer

----------------------
Enter Employee ID: 30
Enter Employee Name: Ajinkya
Enter Department of employee: IT
Enter Designation Of Employee:Backend dev

----------------------
Enter Employee ID: 5
Enter Employee Name: nirav
Enter Department of employee: IT
Enter Designation Of Employee:Manager

=======================================
----------------
Employee ID: 10
Employee Name: Pradip
Employe Department: IT
Employee Designation:senior Developer

----------------
Employee ID: 7
Employee Name: Lakshya
Employee Department: IT
Employee Designation: junior Developer

----------------
Employee ID: 30
Employee Name: Ajinkya
Employee Department: IT

Employee Designation: Backend dev


-----------------

Employee ID: 5

Employee Name: Nirav

Employee Department: IT

Employee Designation: Manager


Details of subordinates of Manager :

-----------------

Employee ID: 10

Employee Name: Pradip

Employoe Department: IT

Employee Designation:senior Developer


-----------------

Employee ID: 7

Employee Name: Lakshya

Employee Department: IT

Employee Designation: junior Developer


-----------------

Employee ID: 30

Employee Name: Ajinkya

Employee Department: IT

Employee Designation: Backend dev


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

****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****

Q(1):A book shop maintains the inventory of books that are being sold at the shop. The list includes details such as author,

title, price, publisher and stock position. Whenever a customer wants a book, the sales person inputs the title and author

and the system searches the list and displays whether it is available or not. If it is not, an appropriate message is displayed.

If it is, then the system displays the book details and requests for the number of copies required. If the requested copies are

available, the total cost of the requested copies is displayed; otherwise the message "Required copies not in stock" is display

Design a system using a class called books with suitable member functions and constructors. Include the following features also :

(a)The number of successful and unsuccessful transaction should be recorded for the purpose of the statistical analysis.

Use static data members to keep count of transactions.

(b) The price of the books should be updated as and when required. Use a private member function to implement this.

****

```
#include<iostream>

#include<string>
```

```
using namespace std;


class books

{


        int stock[20];

        float price[20];

        string author[20],title[20],publisher[20];


public:

        static int un_transaction,sc_transaction,limit;

        books()

        {

                stock[0]=0;

                price[0]=0.0;

                author[0]="";

                title[0]="";

                publisher[0]="";

        }


        void input_books()

        {

                int i,temp_limit;

                cout<<"\n\nEnter Total Number of books you want to enter: ";

                cin>>temp_limit;

                for(i=limit;i<temp_limit+limit;i++)

                {

                        cout<<"\n****************************\n";

                        cout<<"Enter Title of the book: ";

                        cin>>title[i];

                        cout<<"Enter author of the book: ";
        cin>>author[i];
        cout<<"Enter publisher of the book: ";
        cin>>publisher[i];
        cout<<"Enter price of the book: ";
        cin>>price[i];
        cout<<"Enter stock of the book: ";
        cin>>stock[i];
    }
    limit+=temp_limit;
}
void search_book(string t_title,string t_author)
{
    int i,flag=0,n;
    for(i=0;i<limit;i++)
    {
        if(title[i].compare(t_title)==0 && author[i].compare(t_author)==0)
        {
            flag=1;
            break;
        }
    }
    if (!flag)
    {
        cout<<"Your requested book is not available!!";
        un_transaction++;
    }
    else
    {
        cout<<"------------Book Details--------------"<<endl;
        cout<<"Book Title: "<<title[i]<<endl;
        cout<<"Author Name: "<<author[i]<<endl;
```

```
                cout<<"Publisher Name: "<<publisher[i]<<endl;
                cout<<"Book price: "<<price[i]<<endl;
                cout<<"Enter the number of copies required: ";
                cin>>n;
                if(n>stock[i])
                {
                        cout<<"Required copies not in stock!!";
                        un_transaction++;
                }
                else
                {
                        cout<<"The cost of the requested copies: "<<price[i]*n;
                        sc_transaction++;
                }
        }

}

void update_price()
{
        int i,flag=0;
        float t_price;
        string t_title,t_author;
        cout<<"Enter tile of the book you want to update price: ";
        cin>>t_title;
        cout<<"Enter author of thge book you want to update price: ";
        cin>>t_author;

        for(i=0;i<limit;i++)
        {
                if(title[i].compare(t_title)==0 && author[i].compare(t_author)==0)
```

```
            {
                flag=1;
                cout<<"Enter new Price to update: ";
                cin>>t_price;
                price[i]=t_price;
                break;
            }
        }
        if (!flag)
        {
            cout<<"Your requested book is not available!!";
        }



    }
};

int books::limit=0;
int books::un_transaction=0;
int books::sc_transaction=0;
void main()
{
    int key;
    books b;
    do{
        cout<<"\n1-input books\n2-Search Book\n3-update price\n4-Exit\nplease choose: ";
        cin>>key;
        if(key==1)
            b.input_books();
        else if (key==2)
        {
            string title,author;
            cout<<"Enter tile of the book you want to search: ";
            cin>>title;
            cout<<"Enter author of thge book you want ot search: ";
            cin>>author;
            b.search_book(title,author);
        }
        else if(key==3)
            b.update_price();
    }while(key!=4);
    cout<<"\nNumber of succefull transaction: "<<books::sc_transaction;
    cout<<"\nNumber of unsuccefull transaction: "<<books::un_transaction;
}
```

****
****

****
****

output:

```
1-input books
2-Search Book
3-update price
4-Exit
please choose: 1


Enter Total Number of books you want to enter: 2

***************************
Enter Title of the book: c++
Enter author of the book: Dey
Enter publisher of the book: Kumar
Enter price of the book: 980

Enter stock of the book: 23


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

Enter Title of the book: Maths

Enter author of the book: Ramanujan

Enter publisher of the book: Golden

Enter price of the book: 780

Enter stock of the book: 10


1-input books

2-Search Book

3-update price

4-Exit

please choose: 2

Enter tile of the book you want to search: c++

Enter author of thge book you want ot search: ramanujan

Your requested book is not available!!

1-input books

2-Search Book

3-update price

4-Exit

please choose: 2

Enter tile of the book you want to search: Maths

Enter author of thge book you want ot search: Ramanujan

------------Book Details--------------

Book Title: Maths

Author Name: Ramanujan

Publisher Name: Golden

Book price: 780

Enter the number of copies required: 2
```

```
The cost of the requested copies: 1560

1-input books

2-Search Book

3-update price

4-Exit

please choose: 3

Enter tile of the book you want to update price: c++

Enter author of thge book you want to update price: Dey

Enter new Price to update: 450


1-input books

2-Search Book

3-update price

4-Exit

please choose: 4


Number of succefull transaction: 1

Number of unsuccefull transaction: 2
```

****
****

****
****

Q(10):Write a program to create class ‘Search’ having data members (int a[ ], x) and define member functions as void input(),

void output(), void search(int position), void add(int value) to display result (Use New and Delete).

****
****

```
#include<iostream>
using namespace std;

class Search
{
    static int x;
    int *a;

public:
    void input()
    {
        a=new int[5];
        x=5;
        cout<<"Enter "<<x<<" numbers: "<<endl;
        for(int i=0;i<x;i++)
        {
            cin>>*(a+i);
        }
    }

    void output()
    {
        cout<<"Your numbers: \n";
        for(int i=0;i<x;i++)
        {
            cout<<*(a+i)<<endl;
        }
    }
    void add(int n)
    {
        int *ptr;
        ptr= new int[x+1];
        for(int i=0;i<x;i++)
        {
            *(ptr+i)=*(a+i);
        }
        delete[] a;
        a=ptr;
        *(a+x)=n;
        x++;
    }

    void search(int n)
    {
        if(n>x)
            cout<<"Invalid Index!!"<<endl;
        else
        {
            cout<<"Number at index "<<n<<" is: ";
            cout<<*(a+n)<<endl;
        }
    }
};

int Search::x=0;
void main()
{
    int key;
    Search s;
    s.input();
    do
    {
```

```
cout<<"1-Add number in array\n2-Search Number\n3-Display Result\n4-exit";
cout<<"\nChoose one option: ";
cin>>key;
if(key==1)
{
        int x;
        int key2;
        do
        {
                cout<<"Enter the number to add: ";
                cin>>x;
                s.add(x);
                cout<<"Enter 1 to add again: ";
                cin>>key2;
        }while(key2==1);
}
else if(key==2)
{
        int key2,x;
        do
        {
                cout<<"Enter index position to search number in array: ";
                cin>>x;
                s.search(x);
                cout<<"Enter 1 to search again: ";
                cin>>key2;
        }while(key2==1);

}
else if(key==3)
        s.output();
        }while(key !=4);
}
```

****
****

****
****

output:

```
Enter 5 numbers:
1 2 3 4 5
1-Add number in array
2-Search Number
3-Display Result
4-exit
Choose one option: 1
Enter the number to add: 6
Enter 1 to add again: 1
Enter the number to add: 7
Enter 1 to add again: 1
Enter the number to add: 8
Enter 1 to add again: 1
Enter the number to add: 9
Enter 1 to add again: 1
Enter the number to add: 10
Enter 1 to add again: 0
1-Add number in array
2-Search Number
3-Display Result
4-exit
Choose one option: 3
Your numbers:
1
```

```
2
3
4
5
6
7
8
9
10
1-Add number in array
2-Search Number
3-Display Result
4-exit
Choose one option: 2
Enter index position to search number in array: 0
Number at index 0 is: 1
Enter 1 to search again: 1
Enter index position to search number in array: 5
Number at index 5 is: 6
Enter 1 to search again: 0
1-Add number in array
2-Search Number
3-Display Result
4-exit
Choose one option: 3
Your numbers:
1
2
3
4
5
```

```
6
7
8
9
10
1-Add number in array
2-Search Number
3-Display Result
4-exit
Choose one option: 1
Enter the number to add: 11
Enter 1 to add again: 2
1-Add number in array
2-Search Number
3-Display Result
4-exit
Choose one option: 3
Your numbers:
1
2
3
4
5
6
7
8
9
10
11
1-Add number in array
2-Search Number
```

```
3-Display Result

4-exit

Choose one option: 4


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

ASSIGNMENT PRACTICAL - 2

****
****

Name : Pradip S Karmakar.

Roll No : 10

Class : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(1): WAP to use binary operator + add two object of class Numbers having num1 and num2 as its data members and display result.

****
****

```
#include<iostream>
using namespace std;

class Addition
{
	int num;
public:
	void getdata()
	{
		cout<<"\n Enter value:";
		cin>>num;
	}
```

```
        void display()
        {
                cout<<"\n Value is:"<<num;
        }

        Addition operator +(Addition tmp)
        {

                tmp.num = tmp.num+ num;
                return tmp;
        }

};

int main()
{
        Addition no1,no2,no3;
        no1.getdata();
        no1.display();
        no2.getdata();
        no2.display();

        no3=no1+no2;
        no3.display();
        return 0;
}
```

****
****

Output:

```
 Enter value:45


 Value is:45

 Enter value:35


 Value is:35

 Value is:80
```

****
****

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(2):   WAP to overload operator * which multiply a number to each element of an array within a class arrayContainer and display the

result.

****
****

```
#include<iostream>

using namespace std;
```

```
class ArrayContainer
{
        int array[30],n;
public:
        void getdata()
        {
        int i=0;

                cout << "\n How Many Number You Want : ";
                cin >> n;
                for( i = 0; i < n; i++)
                {
                        cout << "\n Enter Number : ";
                        cin >> array[i];
                }

        }
        void display()
-       {
                int i = 0;
                for( i = 0; i < n; i++)
                {
                        cout << "\n Value " << i+1 << " : " << array[i];
                }
        }



        void operator *(int val)
        {
                int res,i=0;
                for( i = 0; i < n; i++)
                {
                  array[i] = array[i] * val;
                }

                for( i = 0; i < n; i++)
                {
                        cout << "\n Value " << i+1 << " : " << array[i];
                }
        }
};

void main()
{
        ArrayContainer array1;



        array1.getdata();
        array1.display();

        array1*(2);
        array1.display();

}
```

****
****

Output:

```
 How Many Number You Want : 2
```

```
Enter number:10

Enter number:20

Value 1 : 10
Value 2 : 20

Value 1 : 20
Value 2 : 40

Value 1 : 20
Value 2 : 40
```

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(3):   WAP to Overload the *, +,-, ==, ! = and = operators for the complex class.

****
****

```
#include<iostream>
```

```
#include<conio.h>
using namespace std;

class Complex
{
        int real;
        float img;

public:
        Complex()
        {
                real=0;
                img=0.0;
        }
        void getdata()
        {
                cout<<"\n Enter real no:";
                cin>>real;
                cout<<"\n Enter img no:";
                cin>>img;
        }
        void display()
        {
                cout<<"\n Real Number is:"<<real;
                cout<<"\n Img Number is:"<<img;
        }

        Complex operator +(Complex tmp)
        {
                Complex c1 ;
                c1.real=real+tmp.real;
```

```
    c1.img=img+tmp.img;

    return (c1);
}
Complex operator -(Complex tmp)
{
    Complex c1 ;
    c1.real=real-tmp.real;
    c1.img=img-tmp.img;

    return (c1);
}
Complex operator *(Complex tmp)
{
    Complex c1;
    c1.real=real*tmp.real;
    c1.img=img*tmp.img;

    return (c1);
}
Complex operator ==(Complex tmp)
{
    Complex c1;

    if(real==tmp.real)
    {
        c1.real=1;
    }
    if(img==tmp.img)
    {
        c1.img=1.0;
```

```
		}

		return (c1);
	}
	Complex operator !=(Complex tmp)
	{
		Complex c1;

		if(real!=tmp.real)
		{
			c1.real=1;
		}
		if(img!=tmp.img)
		{
			c1.img=1.0;
		}

		return (c1);
	}
	void operator =(Complex tmp)
	{

		real=tmp.real;
		img=tmp.img;
	}
};

void main()
{
	Complex c1,c2,c3;
```

```
c1.getdata();
cout<<"\n ------------";
cout<<"\n object 1:";
cout<<"\n ------------";
c1.display();

c2.getdata();
cout<<"\n ------------";
cout<<"\n object 2:";
cout<<"\n ------------";
c2.display();



cout<<"\n ------------";
cout<<"\n Addition:";
cout<<"\n ------------";
c3=c1+c2;
c3.display();

cout<<"\n ------------";
cout<<"\n Subtraction:";
cout<<"\n ------------";
c3=c1-c2;
c3.display();

cout<<"\n ------------";
cout<<"\n Multiplication:";
cout<<"\n ------------";
c3=c1*c2;
c3.display();
```

```
        cout<<"\n ------------";
        cout<<"\n Equal:";
        cout<<"\n ------------";
        c3=c1==c2;
        c3.display();

        cout<<"\n ------------";
        cout<<"\n Not Equal:";
        cout<<"\n ------------";

        c3=c1!=c2;
        c3.display();

        cout<<"\n ------------";
        cout<<"\n Assignment:";
        cout<<"\n ------------";

        c1=c2;
        cout<<"\n ------------";
        cout<<"\n object 1:";
        cout<<"\n ------------";
        c1.display();
        cout<<"\n ------------";
        cout<<"\n object 2:";
        cout<<"\n ------------";

        c2.display();
        getch();

}
```

****
****

Output:

```
 Enter real no:10

 Enter img no:20.10

 ------------
 object 1:
 ------------
 Real Number is:10
 Img Number is:20.1

 Enter real no:40

 Enter img no:20.10

 ------------
 object 2:
 ------------
 Real Number is:40
 Img Number is:20.1
 ------------
 Addition:
 ------------
 Real Number is:50
 Img Number is:40.2
 ------------
```

```
Subtraction:
------------
Real Number is:-30
Img Number is:0
------------
Multiplication:
------------
Real Number is:400
Img Number is:404.01
------------
Equal:
------------
Real Number is:0
Img Number is:1
------------
Not Equal:
------------
Real Number is:1
Img Number is:0
------------
Assignment:
------------
------------
object 1:
------------
Real Number is:40
Img Number is:20.1
------------
object 2:
------------
Real Number is:40
```

Img Number is:20.1

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

Name : Pradip S Karmakar.

Roll No : 10

Class : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(4): WAP to define an object m1 of matrix class, use m1<<cout.

****
****

```
#include<iostream>

#include<string>

#include<conio.h>

using namespace std;


class Matrix

{

        int matrix[10][10];


public:

        Matrix(){}
```

```
ostream & operator <<(ostream & out)
{
        for(int i=0;i<2;i++)
        {
                for(int j=0;j<2;j++)
                {
                        out<<matrix[i][j]<<" ";


                }
                printf("\n");
        }
        return out;
}

istream & operator >>(istream & in)
{
        for(int i=0;i<2;i++)
        {
                for(int j=0;j<2;j++)
                {
                        cout<<"\n Enter Value:";
                        in>>matrix[i][j];
                }
        }
        return in;
}

};
void main()
{
        Matrix m1;
```

```
            cout<<"\n Enter Matrix:";
            m1>>cin;
            cout<<"\n matrix is:";
            m1<<cout;
            getch();
}
```

****
****

Output:

```
 Enter Matrix:
 Enter Value:1


 Enter Value:2


 Enter Value:3


 Enter Value:4


 matrix is:
 1 2
 3 4
```

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(5):   WAP to define a matrix class and overload the * operator to multiply a number with matrix.

****
****

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

class Matrix
{
    int matrix[10][10];

public:
    Matrix(){}

    void input()
    {
        for(int i=0;i<2;i++)
        {
            for(int j=0;j<2;j++)
            {
                cout<<"\n Enter Value:";
                cin>>matrix[i][j];
            }
        }

}
void display()
{
        for(int i=0;i<2;i++)
        {
                for(int j=0;j<2;j++)
                {
                        cout<<"\n Value is:"<<matrix[i][j];

                }
        }
}


Matrix friend operator *(Matrix m4,int mul)
{

        for(int i=0;i<2;i++)
        {
                for(int j=0;j<2;j++)
                {
                        m4.matrix[i][j]=mul * m4.matrix[i][j];

                }
        }
        return m4;

}
```

```
        Matrix friend operator *(int mul,Matrix m4)
        {

                for(int i=0;i<2;i++)
                {
                        for(int j=0;j<2;j++)
                        {
                                m4.matrix[i][j]=mul * m4.matrix[i][j];

                        }
                }
                return m4;

        }

};



void main()
{
        Matrix m1,m2,m3;
        cout<<"\n Enter Value for 2*2 matrix";
        m1.input();
        m1.display();
        cout<<"\n ----------";
        cout<<"\n Matrix*5";
        cout<<"\n ----------";
        m2=m1*5;
        m2.display();

        cout<<"\n ----------";
        cout<<"\n 5*Matrix";
        cout<<"\n ----------";
        m3=5*m1;
        m3.display();
        getch();
}
```

****
****

Output:

```
 Enter Value for 2*2 matrix
 Enter Value:1


 Enter Value:2


 Enter Value:3


 Enter Value:4


 Value is:1
 Value is:2
 Value is:3
 Value is:4
 ----------
 Matrix*5
 ----------
 Value is:5
 Value is:10
 Value is:15
```

```
Value is:20

----------

5*Matrix

----------

Value is:5

Value is:10

Value is:15

Value is:20
```

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

Name : Pradip S Karmakar.

Roll No : 10

Class : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(6): WAP to define a class Date with properties int month; int day; int year;

overload the following operators.

5.1) + operator [a+b] (a is of date type and b is an integer), use the assumption that all years all years have 360 days and

months 30 days.

5.2) – operator [a-b(same as above)]

5.3) = operator

5.4) <,<=,>,>=

5.5) ++,--[post and pre both]

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


#include<iostream>
#include<string>
#include<conio.h>
using namespace std;

class Date
{
	int month,day,year;
public:
	void getdate()
	{
		cout<<"\n Enter Date:";
		cin>>day>>month>>year;
	}
	void display()
	{
		cout<<"\n Date is:"<<day<<'/'<<month<<'/'<<year;
	}

	void operator +(int tday)
	{
		day=day + tday;

		while(day>30)
		{
			day=day-30;
			month++;
			if(month>12)
```

```
                {
                        month=1;
                        year++;
                }
        }
}

void operator -(int tday)
{
        day=day - tday;

        while(day<0)
        {
                month--;
                if(month==0)
                {
                        month=12;
                        year--;
                }
                day=30+day;
        }
}

int operator >(Date d2)
{
        int flag=0;

        if(year>d2.year)
        {
                flag=1;
```

```
    }
    else if(year==d2.year)
    {
        if(month>d2.month)
        {
            flag=1;
        }
        else if(month==d2.month)
        {
            if(day>d2.day)
            {
                flag=1;
            }
            else
            {
                flag=0;
            }
        }
    }
    return flag;
}

int operator <(Date d2)
{
    int flag=0;

    if(year<d2.year)
    {
        flag=1;

    }
```

```
	else if(year==d2.year)
	{
		if(month<d2.month)
		{
			flag=1;
		}
		else if(month==d2.month)
		{
			if(day<d2.day)
			{
				flag=1;
			}
			else
			{
				flag=0;
			}
		}
	}
	return flag;
}
int operator >=(Date d2)
{
	int flag=0;

	if(year>d2.year)
	{
		flag=1;


	}
	else if(year==d2.year)
	{
        if(month>d2.month)
        {
            flag=1;
        }
        else if(month==d2.month)
        {
            if(day>=d2.day)
            {
                flag=1;
            }
            else
            {
                flag=0;
            }
        }
    }
    return flag;
}

int operator <=(Date d2)
{
    int flag=0;

    if(year<d2.year)
    {
        flag=1;

    }
    else if(year==d2.year)
    {
        if(month<d2.month)
        {
            flag=1;
        }
        else if(month==d2.month)
        {
            if(day<=d2.day)
            {
                flag=1;
            }
            else
            {
                flag=0;
            }
        }
    }
    return flag;
}
void operator ++()
{
    Date d1;
    if(day==30)
    {
        day=1;
        if(month==12)
        {
            month=1;
            year++;
        }
        else
            month++;
    }
```

```
        else
        {
            day++;
        }

    }

    void operator --()
    {

        if(day==1)
        {
            day=30;
            if(month==1)
            {
                month=12;
                year--;
            }
            else
                month--;
        }
        else
        {
            day--;
        }

    }
};

void main()
{
```

```
int day,res,flag=0,ch;
char ch2='n';

Date d1,d2;

cout<<"\n----------------------";
cout<<"\n 1.Addition(+) of Days";
cout<<"\n 2.Subtraction(-) of Days";
cout<<"\n 3.check two date (>)";
cout<<"\n 4.check two date (>=)";
cout<<"\n 5.check two date (<)";
cout<<"\n 6.check two date (<=)";
cout<<"\n 7.Increment date (++)";
cout<<"\n 8.Decrement date (--)";
cout<<"\n----------------------";

do
{

cout<<"\n\n Enter Your choice:";
cin>>ch;

switch(ch)
{
        case 1:  d1.getdate();
                        cout<<"\n Add Number of days:";
                        cin>>day;

                        d1+(day);
                        cout<<"\n\n Date After Adding "<<day<<"into date ";
                        cout<<"\n-------------------------------------";
```

```
                d1.display();
                break;

case 2: d1.getdate();
                cout<<"\n Subtract Number of days:";
                cin>>day;

                d1-(day);
                cout<<"\n\n Date After Subtracting "<<day<<"into date ";
                cout<<"\n---------------------------------------";
                d1.display();
                break;

case 3: d1.getdate();
                d2.getdate();

                res=d1>d2;
                if(res==1)
                {
                        cout<<"Date1 is greter.";
                }
                else
                {
                        cout<<"Date1 is not greter.";
                }
                break;

case 4: d1.getdate();
                d2.getdate();

                res=d1>=d2;
                if(res==1)

                {

                        cout<<"Date1 is greter.";

                }

                else

                {

                        cout<<"Date1 is not greter.";

                }

                break;


case 5: d1.getdate();

                d2.getdate();


                res=d1<d2;

                if(res==1)

                {

                        cout<<"Date2 is greter.";

                }

                else

                {

                        cout<<"Date2 is not greter.";

                }

                break;


case 6: d1.getdate();

                d2.getdate();


                res=d1<=d2;

                if(res==1)

                {

                        cout<<"Date is greter.";
                }
                else
                {
                    cout<<"Date is not greter.";
                }
                break;

        case 7: d1.getdate();
                d1.display();
                ++d1;
                break;

        case 8: d1.getdate();
                d1.display();
                --d1;
                break;

    }

    cout<<"\n Do You Want to continue:";
    cin>>ch2;

    }while(ch2=='y' || ch2=='y');
    getch();
}
```

****
****

Output:

```
 ----------------------
 1.Addition(+) of Days
 2.Subtraction(-) of Days
 3.check two date (>)
 4.check two date (>=)
 5.check two date (<)
 6.check two date (<=)
 7.Increment date (++)
 8.Decrement date (--)
----------------------

 Enter Your choice:1

 Enter Date:5 2 2019

 Add Number of days:375


 Date After Adding 375into date
---------------------------------------
 Date is:20/2/2020
 Do You Want to continue:y


 Enter Your choice:2

 Enter Date:5 1 2019

 Subtract Number of days:25
 Date After Subtracting 25into date
--------------------------------------
 Date is:10/12/2018
 Do You Want to continue:y


 Enter Your choice:3

 Enter Date:5 2 2018

 Enter Date:1 2 2018
Date1 is greter.
 Do You Want to continue:y


 Enter Your choice:4

 Enter Date: 4 5 2019

 Enter Date:4 5 2019
Date1 is greter.
 Do You Want to continue:y


 Enter Your choice:5

 Enter Date:5 4 2019

 Enter Date:1 2 2018
Date2 is not greter.
 Do You Want to continue:y
```

```
Enter Your choice:7

Enter Date:30 12 2019

Date is:1/1/2020
Do You Want to continue:y


Enter Your choice:8

Enter Date:1 1 2019

Date is:30/12/2018
Do You Want to continue:n
```

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(7):   WAP to define a class Time with properties int hour; int minute; int second;

overload the following operators.

6.1) + operator [a+b] (a is of time type and b is an integer)

6.2) – operator [a-b(same as above)]

6.3) = operator

6.4) <,<=,>,>=

6.5) ++,--[post and pre both]

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

#include<iostream>
#include<string>
#include<conio.h>
using namespace std;

class Time
{
	int hh,mm,ss;
public:

	void getTime()
	{
		cout<<"Enter Hours:";
		cin>>hh;
		cout<<"Enter Minute:";
		cin>>mm;
		cout<<"Enter Secound:";
		cin>>ss;

	}
	void display()
	{
        cout<<"\n";
        cout<<"\n"<<hh<<":"<<mm<<":"<<ss;

}

void operator +(int tmin)
{

        mm+=tmin;

        while(mm>=60)
        {
                mm=mm-60;
                hh++;

                if(hh>24)
                {
                        hh=1;
                }
        }
        if(hh>24)
        {
                hh=1;
        }

}

void operator -(int tmin)
{

        mm=mm-tmin;
        while(mm<0)
        {

                hh--;

                if(hh==0)
                {
                        hh=24;
                }
                mm=mm+60;
        }

}

int operator >(Time t2)
{
        int flag=0;

        if(hh>t2.hh)
        {
                flag=1;

        }
        else if(hh==t2.hh)
        {
                if(mm>t2.mm)
                {
                        flag=1;
                }
                else if(mm==t2.mm)
        {
            if(ss>t2.ss)
            {
                flag=1;
            }
            else
            {
                flag=0;
            }
        }
    }
    return flag;
}

int operator <(Time t2)
{
    int flag=0;

    if(hh<t2.hh)
    {
        flag=1;

    }
    else if(hh==t2.hh)
    {
        if(mm<t2.mm)
        {
            flag=1;
        }
        else if(mm==t2.mm)
        {
```

```
                        if(ss<t2.ss)
                        {
                                flag=1;
                        }
                        else
                        {
                                flag=0;
                        }
                }
        }
        return flag;
}

int operator >=(Time t2)
{
        int flag=0;

        if(hh>t2.hh)
        {
                flag=1;

        }
        else if(hh==t2.hh)
        {
                if(mm>t2.mm)
                {
                        flag=1;
                }
                else if(mm==t2.mm)
                {
                        if(ss>=t2.ss)
```

```
                    {
                              flag=1;
                    }
                    else
                    {
                              flag=0;
                    }
          }
     }
     return flag;
}


int operator <=(Time t2)
{
     int flag=0;

     if(hh<t2.hh)
     {
          flag=1;


     }
     else if(hh==t2.hh)
     {
          if(mm<t2.mm)
          {
               flag=1;
          }
          else if(mm==t2.mm)
          {
               if(ss<=t2.ss)
```

```
                        {
                                flag=1;
                        }
                        else
                        {
                                flag=0;
                        }
                }
        }
        return flag;
}


void operator ++()
{

        if(mm==60)
        {
                mm=1;
                hh++;           }
        else
        {
                mm++;
        }

}


void operator --()
{
        if(mm==0)
        {
            mm=60;
            hh--;            }
        else
        {
            mm--;
        }

    }


};


void main()
{
    int min,res,flag=0,ch;
    char ch2='n';

    Time t1,t2;

    cout<<"\n----------------------";
    cout<<"\n 1.Addition(+) of Days";
    cout<<"\n 2.Subtraction(-) of Days";
    cout<<"\n 3.chack two date (>)";
    cout<<"\n 4.chack two date (>=)";
    cout<<"\n 5.chack two date (<)";
    cout<<"\n 6.chack two date (<=)";
    cout<<"\n 7.Increment date (++)";
    cout<<"\n 8.Decrement date (--)";
```

```
cout<<"\n---------------------";


do

{


cout<<"\n\n Enter Your choice:";

cin>>ch;


switch(ch)

{

        case 1:  t1.getTime();

                        cout<<"\n Add Number of min:";

                        cin>>min;


                        t1+(min);

                        cout<<"\n\n Time After Adding "<<min<<"into time ";

                        cout<<"\n-----------------------------------";

                        t1.display();

                        break;


        case 2: t1.getTime();

                        cout<<"\n Subtract Number of min:";

                        cin>>min;


                        t1-(min);

                        cout<<"\n\n Time After Adding "<<min<<"into time ";

                        cout<<"\n-----------------------------------";

                        t1.display();

                        break;


        case 3: t1.getTime();
```

```
            t2.getTime();

            res=t1>t2;
            if(res==1)
            {
                  cout<<"Time1 is greter.";
            }
            else
            {
                  cout<<"Time1 is not greter.";
            }
            break;

case 4: t1.getTime();
            t2.getTime();

            res=t1>=t2;

            if(res==1)
            {
                  cout<<"Time1 is greter.";
            }
            else
            {
                  cout<<"Time1 is not greter.";
            }
            break;

case 5: t1.getTime();
            t2.getTime();
```

```
                    res=t1<t2;
                    if(res==1)
                    {
                              cout<<"Time2 is greter.";
                    }
                    else
                    {
                              cout<<"Time2 is not greter.";
                    }
                    break;

case 6: t1.getTime();
                    t2.getTime();

                    res=t1<=t2;
                    if(res==1)
                    {
                              cout<<"Time2 is greter.";
                    }
                    else
                    {
                              cout<<"Time2 is not greter.";
                    }
                    break;

case 7: t1.getTime();
                    t1++;
                    t1.display();
                    break;

case 8: t1.getTime();
```

```
                                        t1--;
                                        t1.display();
                                        break;

            }

            cout<<"\n Do You Want to continue:";
            cin>>ch2;

            }while(ch2=='y' || ch2=='y');
getch();
}
```

****

Output:

```
 ----------------------
 1.Addition(+) of Days
 2.Subtraction(-) of Days
 3.chack two date (>)
 4.chack two date (>=)
 5.chack two date (<)
 6.chack two date (<=)
 7.Increment date (++)
 8.Decrement date (--)
----------------------

Enter Your choice:1
Enter Hours:12
Enter Minute:45

Enter Secound:10


 Add Number of min:110



 Time After Adding 110into time

---------------------------------------


14:35:10

 Do You Want to continue:y



Enter Your choice:2

Enter Hours:12

Enter Minute:00

Enter Secound:10


 Subtract Number of min:60



 Time After Adding 60into time

---------------------------------------


11:0:10

 Do You Want to continue:y



Enter Your choice:3

Enter Hours:10

Enter Minute:45
Enter Secound:07

Enter Hours:10

Enter Minute:45

Enter Secound:07

Time1 is not greter.

 Do You Want to continue:y



Enter Your choice:4

Enter Hours:23

Enter Minute:21

Enter Secound:12

Enter Hours:23

Enter Minute:21

Enter Secound:12

Time1 is greter.

 Do You Want to continue:y



 Enter Your choice:5

Enter Hours:10

Enter Minute:10

Enter Secound:10

Enter Hours:10

Enter Minute:9

Enter Secound:45

Time2 is not greter.

 Do You Want to continue:6
```

```
Enter Your choice:6

Enter Hours:14

Enter Minute:24

Enter Secound:60

Enter Hours:15

Enter Minute:10

Enter Secound:14

Time2 is greter.

 Do You Want to continue:y



Enter Your choice:7

Enter Hours:25

Enter Minute:56

Enter Secound:50



25:57:50

 Do You Want to continue:y



Enter Your choice:8

Enter Hours:60

Enter Minute:0

Enter Secound:45



59:60:45

 Do You Want to continue:n
```

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(8):   Write a menu driven program that can perform the following functions on strings. (Use overloaded operators where possible).

(Do not use predefined string class)

1. Compare two strings for equality (== operator)

2. Check whether first string is smaller than the second (<= operator)

3. Copy the string to another

4. Extract a character from the string (Overload [])

5. Reverse the string

6. Concatenate two strings (+ operator)

****
****

```
#include<iostream>

#include<conio.h>

using namespace std;


class String

{
```

```
// string str;
 char str[20];
 public:
    void getdata()
    {
        cout<<"\n Enter String:";
        cin.get(str,20);
    }

    void display()
    {
         int i=0;
        cout<<"\n";
        while(str[i]!='\0')
        {
            cout<<str[i];
            i++;
        }

    }

    bool operator ==(String s2)
    {
         int i=0,j=0,flag=1;

        while(str[i]!='\0' && s2.str[j]!='\0' && str[i]==s2.str[j])
        {
                    i++;
                    j++;
        }
        if(str[i]=='\0' && s2.str[j]=='\0')
```

```
            {
                        return true;
            }
            else
            {
                        return false;

            }



}

bool operator <=(String s2)
{
             int i=0,j=0,flag=1,res;
             cout<<str;
            while(str[i]!='\0' && s2.str[j]!='\0' && str[i]==s2.str[j])
            {
                                    i++;
                                    j++;
            }
            if(str[i]=='\0' && s2.str[j]=='\0')
            {
                        return false;
            }
            else
            {
                        res=str[i]-s2.str[j];
                        if(res<0)
                                    return true;
                        else
            return false;
    }



}

void operator =(String s2)
{
    int i=0;

    while(s2.str[i]!='\0')
    {
        str[i]=s2.str[i];
        i++;
    }
    str[i]='\0';

}
char operator [](int index)
{
    return str[index];
}

String operator +(String s2)
{
     String str3;
     int i=0,j=0;

    while(str[i]!='\0')
    {
        str3.str[i]=str[i];
```

```
                i++;
        }
        while(s2.str[j]!='\0')
        {
                str3.str[i]=s2.str[j];
                i++;
                j++;
        }
        str3.str[i]='\0';
        return str3;
}

int length(char str[20])
{

        int len=0,i=0;
        while(str[i]!='\0')
        {
                i++;
                len++;
        }

        return len;
}
void reverse()
{
        int i=0,j=0,len,temp;
        len=length(str);

        i=0;
        j=len-1;
```

```
            while(i!=(len/2))
            {
                temp=str[i];
                str[i]=str[j];
                str[j]=temp;
                i++;
                j--;
            }

        }

};

void main()
{
    char ch,cont='n';
    String s1,s2,s3;
    bool res;
    int index,choice;
    s1.getdata();
    flushall();
    s2.getdata();
    cout<<"\n--------------------------------------";
    cout<<"\n 1. Compare two strings for equality";
    cout<<"\n 2. Check whether first string is smaller than the second";
    cout<<"\n 3. Copy the string to another";
    cout<<"\n 4. Extract a character from the string";
    cout<<"\n 5. Reverse the string";
    cout<<"\n 6. Concatenate two strings";
    cout<<"\n--------------------------------------";
    do
```

```
{
        cout<<"\n Enter Your Choice:";
        cin>>choice;

switch(choice)
{
        case 1:

                        if(s1==s2)
                                cout<<"\n Both Strings Are Equal.";
                        else
                                cout<<"\n Both Strings Are Not Equal.";
                        break;
        case 2:
                        if(s1<=s2)
                                cout<<"\n First String is Smaller than Secound.";
                        else
                                cout<<"\n First String is Not Smaller than Secound.";
                        break;
        case 3:

                        s2=s1;
                        cout<<"\n-------------";
                        cout<<"\n Frist String";
                        cout<<"\n-------------\n";
                        s1.display();
                        cout<<"\n-------------";
                        cout<<"\n Secound String";
                        cout<<"\n-------------\n";
                        s2.display();
                        break;
```

```
                case 4:

                        s1.reverse();
                        cout<<"\n---------------";
                        cout<<"\n Reverse String";
                        cout<<"\n---------------\n";
                        s1.display();
                        break;
                case 5:
                        cout<<"\n Enter index:";
                        cin>>index;
                        ch=s1[index];
                        cout<<"character is:"<<ch;
                        break;
                case 6:
                        s3=s1+s2;
                        cout<<"\n --------";
                        cout<<"\n String:";
                        cout<<"\n --------\n";
                        s3.display();
                        break;
        }
                cout<<"\n\n Do You Want to Continue(y/n)";
                cin>>cont;
        }while(cont=='Y' || cont=='y');

        getch();
}
```

****
****

Output:

```
 Enter String:Rollwala

 Enter String:DCSGU

-------------------------------------
 1. Compare two strings for equality
 2. Check whether first string is smaller than the second
 3. Copy the string to another
 4. Extract a character from the string
 5. Reverse the string
 6. Concatenate two strings
-------------------------------------
 Enter Your Choice:1

 Both Strings Are Not Equal.

 Do You Want to Continue(y/n)y

 Enter Your Choice:2
Rollwala
 First String is Not Smaller than Secound.

 Do You Want to Continue(y/n)y

 Enter Your Choice:3

-------------
 Frist String
```

```
-------------


Rollwala

-------------

 Secound String

-------------


Rollwala


 Do You Want to Continue(y/n)y


 Enter Your Choice:4


---------------

 Reverse String

---------------


alawlloR


 Do You Want to Continue(y/n)y


 Enter Your Choice:5


 Enter index:2

character is:a


 Do You Want to Continue(y/n)y


 Enter Your Choice:6


 --------
 String:
 --------

alawlloRRollwala

 Do You Want to Continue(y/n)n
```

****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****

Q(9):   Design a manipulator to provide the following output specifications for printing float values

(i) 5 column width

(ii) Right justified

(iii) 2 digits precision

(iv) Filling unused spaces with +

****

```
#include<stdio.h>
```

```
#include<conio.h>

#include<iostream>

#include<iomanip>

#include<string>

using namespace std;


ostream & convert(ostream & pout)

{

                pout<<setw(5)<<setprecision(5)<<setfill('+')<<setiosflags(ios::right);

                return pout;

}


void main()

{


        float amt;

        cout<<"\n Enter Amount:";

        cin>>amt;

        cout<<convert;

        cout<<amt;


        getch();

}
```

****
****

Output:

```
 Enter Amount:1.2

        ++1.2
```

****
****

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(10): Define a class marksheet. The class should contain a function PrintMarkSheet such that it prints the marksheet of a given student with three subject names and five marks for each subject. Define manipulators for displaying headings and footnotes. The function should display marksheet with respective headings and class. The marks should be aligned under the headings (Use either ios functions or manipulators).

****
****

```
#include<stdio.h>
#include<conio.h>
#include<iostream>
#include<iomanip>
#include<string>
using namespace std;
```

```
namespace A
{
        ostream & PrintLine(ostream & pout)
        {
            pout<<"\n--------------------------------------------------------------------------------------------------"<<endl;
            return pout;
        }
}

class Marksheet
{

    string sub[3],mark_type[3],course_name;
    int marks[3][3];
public:
    Marksheet()
    {
        mark_type[0] = "Term Work";
        mark_type[1] = "Sessional-1";
        mark_type[2] = "Sessional-2";
        sub[0]="Advanced Programming";
        sub[1]="Object Oriented Programming";
        sub[2]="Database Management System";
    }
    void Input()
    {
        int j=0,i=0;


        cout<<"\n Enter Course Name:";
        getline(cin,course_name);

        for(i=0;i<3;i++)
        {
                cout<<"\n" <<sub[i];
                cout<<"\n------------------------";
                for(j=0;j<3;j++)
                {
                        cout<<"\n Enter " <<mark_type[j]<<" Marks:";
                        cin>>marks[i][j];
                }
        }

}
void Display()
{
        int j=0,i=0;

        cout<<"\n Course Name :"<<course_name;

        for(i=0;i<3;i++)
        {
                cout<<"\n" <<sub[i];
                cout<<"\n------------------------";
                for(j=0;j<3;j++)
                {
                        cout<<"\n Enter " <<mark_type[j]<<" Marks :"<<marks[i][j];
                }
        }
}
```

```
void PrintMarksheet()
{
        int i,j,sum[3],tot=0;

        cout.width(60);
        cout.fill(' ');
        cout<<"GUJRAT UNIVERSITY";

        cout<<A::PrintLine;

        //cout<<setw(20);
        cout.width(70);
        cout.fill(' ');
        cout.setf(ios::internal);

        cout<<course_name;
        cout.width(30);
        cout.fill(' ');
        cout<<"|";
        cout<<A::PrintLine;

        cout<<setw(30)<<"Subject Name"<<setw(20)<<setiosflags(ios::right)<<setfill('
')<<mark_type[0]<<setw(15)<<setfill(' ')<<mark_type[1]<<setw(20)<<setfill('
')<<mark_type[2]<<setw(10)<<setfill(' ')<<"Total"<<setw(5)<<setfill(' ')<<"|";
        cout<<A::PrintLine;
        for(i=0;i<3;i++)
        {
                for(j=0;j<1;j++)
                {
                        sum[i]=marks[i][j]+marks[i][j+1]+marks[i][j+2];
```

```
    cout<<"\n"<<setw(30)<<sub[i]<<setw(20)<<marks[i][j]<<setw(15)<<marks[i][j+1]<<setw(20)<<marks[i][j+2]<<setw(10)<<sum[i]<<setw(5)<<setfill(' ')<<"|";
            }
        }

        for(i=0;i<3;i++)
        {
            tot=tot+sum[i];
        }
        cout<<A::PrintLine;
        cout<<setw(40)<<setfill(' ')<<"Percentage is:"<<((tot*100)/900)<<setw(55)<<setfill(' ')<<tot<<setw(2)<<setfill(' ')<<"|";;
        cout<<A::PrintLine;

    }

};



void main()
{
    Marksheet m1;
    m1.Input();
    //m1.Display();
    m1.PrintMarksheet();
    getch();
}
```

****
****

Output:

```
 Enter Course Name:MASTER OF COMPUTER APPLICATIONS

Advanced Programming
------------------------
 Enter Term Work Marks:45

 Enter Sessional-1 Marks:78

 Enter Sessional-2 Marks:56

Object Oriented Programming
------------------------
 Enter Term Work Marks:66

 Enter Sessional-1 Marks:67

 Enter Sessional-2 Marks:89

Database Management System
------------------------
 Enter Term Work Marks:57

 Enter Sessional-1 Marks:54

 Enter Sessional-2 Marks:56
                                GUJRAT UNIVERSITY
----------------------------------------------------------------------------------------------------
                             MASTER OF COMPUTER APPLICATIONS                         |
----------------------------------------------------------------------------------------------------
```

```
            Subject Name          Term Work    Sessional-1        Sessional-2     Total    |
------------------------------------------------------------------------------------------------


       Advanced Programming                45           78                56       179    |
  Object Oriented Programming                66           67                89       222    |
   Database Management System                57           54                56       167    |
------------------------------------------------------------------------------------------------
                  Percentage is:63                                       568    |
------------------------------------------------------------------------------------------------


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

ASSIGNMENT PRACTICAL - 3

****
****

Name : Pradip S Karmakar.

Roll No : 10

Class : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(1): Write a program to generate templates function for swapping values of variables and show its use with integer, float and character type of data as

input.

****
****

```
#include<iostream>

#include<conio.h>

using namespace std;


template <typename T>


void Swap(T *a,T *b)

{

        T temp;

        temp=*a;

        *a=*b;

        *b=temp;
```

```
}

void main()
{
	int a,b;
	float c,d;
	char ch1,ch2;

	cout<<"\n Enter A:";
	cin>>a;
	cout<<"\n Enter B:";
	cin>>b;
	cout<<"\n a is:"<<a<<"\n b is:"<<b;
	cout<<"\n-------------------------";
	cout<<"\n After Swap";
	cout<<"\n-------------------------";
	Swap<int>(&a,&b);
	cout<<"\n a is:"<<a<<"\n b is:"<<b;

	cout<<"\n\n Enter C:";
	cin>>c;
	cout<<"\n Enter D:";
	cin>>d;
	cout<<"\n C is:"<<c<<"\n D is:"<<d;
	cout<<"\n-------------------------";
	cout<<"\n After Swap";
	cout<<"\n-------------------------";
	Swap<float>(&c,&d);
	cout<<"\n C is:"<<c<<"\n D is:"<<d;

	cout<<"\n\n Enter character1:";
```

```
            cin>>ch1;

            cout<<"\n Enter character2:";

            cin>>ch2;

            cout<<"\n ch1 is:"<<ch1<<"\n ch2 is:"<<ch2;

            cout<<"\n-------------------------";

            cout<<"\n After Swap";

            cout<<"\n-------------------------";

            Swap<char>(&ch1,&ch2);

            cout<<"\n ch1 is:"<<ch1<<"\n ch2 is:"<<ch2;



            getch();

}
```

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

Output :

```
 Enter A : 24


 Enter B : 12


 a is : 24

 b is : 12

-------------------------

 After Swap

-------------------------

 a is : 12

 b is : 24
 Enter C:27.50


 Enter D:19.5


 C is:27.5

 D is:19.5

-------------------------

 After Swap

-------------------------

 C is:19.5

 D is:27.5


 Enter character1:A


 Enter character2:B


 ch1 is:A

 ch2 is:B

-------------------------

 After Swap

-------------------------

 ch1 is:B

 ch2 is:A
```

****
****

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****

Q(2): Write an object oriented program to implement a generic Stack. Incorporate all the possible operation on Stack in the program.

****

```
#include<iostream>
using namespace std;

template <typename T>

class Stack
{
	T stack[5];
	int top,i,res;
public:

	Stack()
	{
		top=-1;
	}
	void Push(T a)
	{
		if(top>5)
		{
			cout<<"\n Stack is Overflow.";
		}
```

```
    else
    {
        top++;
        stack[top]=a;
    }
}

void Pop()
{
    if(top==-1)
    {
        cout<<"\n Stack is Underflow or Empty.";
    }
    else
    {
        stack[top]=NULL;
        top--;
        cout<<"\n Deleted Successfully.";
    }
}
void Display()
{
    if(top==-1)
    {
        cout<<"\n Stack is empty.";
    }
    else
    {
        for(i=0;i<=top;i++)
        {
            cout<<"\n val is:"<<stack[i];
```

```
                }
            }
        }
};

void main()
{
    Stack <int>s1;

    int ch,val,res;
    char choice='n';

    cout<<"\n--------------------";
    cout<<"\n 1.Push";
    cout<<"\n 2.Pop";
    cout<<"\n 3.Display";
    cout<<"\n--------------------";

    do
    {
        cout<<"\n Enter Your Choice:";
        cin>>ch;

        switch(ch)
        {
            case 1:
                    cout<<"\n Enter Value:";
                    cin>>val;
                    s1.Push(val);
                    break;
            case 2:
                    s1.Pop();
                    break;
            case 3:
                    s1.Display();
                    break;
        }
        cout<<"\n Do You Want to continue:";
        cin>>choice;

    }while(choice=='Y' || choice=='y');
}
```

****
****

Output :

```
--------------------
 1.Push
 2.Pop
 3.Display
--------------------
 Enter Your Choice:1

 Enter Value:10

 Do You Want to continue:y

 Enter Your Choice:1

 Enter Value:20
```

```
Do You Want to continue:y

Enter Your Choice:1

Enter Value:30

Do You Want to continue:y

Enter Your Choice:3

val is:10
val is:20
val is:30
Do You Want to continue:y

Enter Your Choice:2

Deleted Successfully.
Do You Want to continue:y

Enter Your Choice:3

val is:10
val is:20
Do You Want to continue:y

Enter Your Choice:2

Deleted Successfully.
Do You Want to continue:y
```

```
 Enter Your Choice:2


 Deleted Successfully.

 Do You Want to continue:y


 Enter Your Choice:3


 Stack is empty.

 Do You Want to continue:y


 Enter Your Choice:2


 Stack is Underflow or Empty.

 Do You Want to continue:n
```

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(3): Write a generic function that will sort a character string, integer and float value. Create a menu with appropriate options and accept the values from

the user.

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


#include<iostream>
#include<conio.h>
#include<string>
using namespace std;

template <typename T>

void Bubble_sort(T arr[])
{
	int i,j,n=5;
	T temp;
	for(i=0;i<n;i++)
	{
		for(j=0;j<n-i-1;j++)
		{
			if(arr[j]>arr[j+1])
			{
				temp=arr[j];
				arr[j]=arr[j+1];
				arr[j+1]=temp;
			}
		}
	}
	cout<<"\n Sorted Completed.";
}

template <typename T>
```

```
void Display(T arr[])
{
        int i;
        cout<<"\n------------------------------";
        for(i=0;i<5;i++)
        {
                cout<<"\n Value is:"<<arr[i];
        }

}

void main()
{
        int arr[5],i;
        float farr[5];
        char carr[10];

        int ch,val,res;
        char choice='n';



        cout<<"\n--------------------";
        cout<<"\n 1.Int";
        cout<<"\n 2.Float";
        cout<<"\n 3.char";
        cout<<"\n--------------------";

        do
        {
                cout<<"\n Enter Your Choice:";
                cin>>ch;
```

```
switch(ch)
{
        case 1:
                        for(i=0;i<5;i++)
                        {
                                cout<<"\n Enter Value:";
                                cin>>arr[i];
                        }

                        Bubble_sort<int>(arr);
                        Display<int>(arr);
                        break;
        case 2:
                        for(i=0;i<5;i++)
                        {
                                cout<<"\n Enter Value:";
                                cin>>farr[i];
                        }

                        Bubble_sort<float>(farr);
                        Display<float>(farr);
                        break;
        case 3:
                        cout<<"Enter String:";
                        cin>>carr;

                        Bubble_sort<char>(carr);
                        Display<char>(carr);
                        break;
        case 4:
```

```
                    break;
        }
        cout<<"\n Do You Want to continue:";
        cin>>choice;

    }while(choice=='Y' || choice=='y');
    getch();
}
```

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

Output:

```
--------------------
 1.Int
 2.Float
 3.char
--------------------
 Enter Your Choice:1

 Enter Value:4

 Enter Value:7

 Enter Value:8

 Enter Value:9

 Enter Value:1
```

```
Sorted Completed.
------------------------------
Value is:1
Value is:4
Value is:7
Value is:8
Value is:9
Do You Want to continue:y

Enter Your Choice:2

Enter Value:4.5

Enter Value:6.32

Enter Value:0.12

Enter Value:2.12

Enter Value:4.5

Sorted Completed.
------------------------------
Value is:0.12
Value is:2.12
Value is:4.5
Value is:4.5
Value is:6.32
Do You Want to continue:y
```

```
 Enter Your Choice:3
Enter String:hello

 Sorted Completed.
------------------------------
 Value is:e
 Value is:h
 Value is:l
 Value is:l
 Value is:o
 Do You Want to continue:n
```

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(4):   Write a template function called find(). This function searches an array for an object. It returns either the index of the matching object (if one is found) or

-1 if no match is found.

****
****

```
#include<iostream>
```

```
#include<string>

#include<conio.h>

using namespace std;



class Student

{

        public:

        int rno;

        string name;

public:

        void addstudent()

        {

                cout<<"\n Enter Roll no:";

                cin>>rno;

                cout<<"\n Enter Name:";

                cin>>name;

        }

        void display()

        {

                cout<<"\n roll no is:"<<rno;

                cout<<"\n Name is:"<<name;

        }


};


template <typename T>


int find(T arrayobj[] ,int rollno,int n)

{

        int flag=0,index=0;
        for(int i=0;i<n;i++)
        {
                if(arrayobj[i].rno==rollno)
                {       flag=1;
                        index=i;
                        break;
                }
        }
        if(flag==0)
                return -1;
        else
                return index;
}

void main()
{
        Student s1[20];
        int rollno,n,index;

        cout<<"\n Enter total value of student data:";
        cin>>n;
        for(int i=0;i<n;i++)
        {
                s1[i].addstudent();
        }

        cout<<"\n Total Student is:";

        cout<<"\n-------------------";
        for(int i=0;i<n;i++)
        {
                s1[i].display();
        }
        cout<<"\n------------------";

        cout<<"\n Enter roll no which you want to find:";
        cin>>rollno;

        index=find<Student>(s1,rollno,n);
        if(index==-1)
                cout<<"\n rollno is not found";
        else
                cout<<"\n rollo is foud";
        getch();
}
```

****
****

Output:

```
 Enter total value of student data:2

 Enter Roll no:1

 Enter Name:sa

 Enter Roll no:2

 Enter Name:mina
 Total Student is:

-------------------

 roll no is:1

 Name is:sa

 roll no is:2

 Name is:mina

-------------------

 Enter roll no which you want to find:4


 rollno is not found
```

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(5):   WAP Implement template sort with a non type size.

****
****

```
#include<iostream>

#include<conio.h>

#include<string>

using namespace std;


template <typename T>
```

```
void Bubble_sort(T arr[],int n) //Non-Type Argument

{

        int i,j;

        T temp;

        for(i=0;i<n;i++)

        {

                for(j=0;j<n-i-1;j++)

                {

                        if(arr[j]>arr[j+1])

                        {

                                temp=arr[j];

                                arr[j]=arr[j+1];

                                arr[j+1]=temp;

                        }

                }

        }

        cout<<"\n Sorted Completed.";

}


template <typename T>


void Display(T arr[])

{

        int i;

        cout<<"\n------------------------------";

        for(i=0;i<5;i++)

        {

                cout<<"\n Value is:"<<arr[i];

        }


}
```

```
void main()
{
    int arr[5],i;
    float farr[5];
    char carr[5];

    int ch,val,res;
    char choice='n';



    cout<<"\n--------------------";
    cout<<"\n 1.Int";
    cout<<"\n 2.Float";
    cout<<"\n 3.char";
    cout<<"\n--------------------";

    do
    {
        cout<<"\n Enter Your Choice:";
        cin>>ch;

        switch(ch)
        {
            case 1:
                    for(i=0;i<5;i++)
                    {
                        cout<<"\n Enter Value:";
                        cin>>arr[i];
                    }
                    Bubble_sort<int>(arr,5);
                    Display<int>(arr);
                    break;
            case 2:
                    for(i=0;i<5;i++)
                    {
                        cout<<"\n Enter Value:";
                        cin>>farr[i];
                    }

                    Bubble_sort<float>(farr,5);
                    Display<float>(farr);
                    break;
            case 3:
                    cout<<"Enter String:";
                    cin>>carr;

                    Bubble_sort<char>(carr,5);
                    Display<char>(carr);
                    break;
            case 4:

                    break;
        }
        cout<<"\n Do You Want to continue:";
        cin>>choice;

    }while(choice=='Y' || choice=='y');
    getch();
}
```

****
****

Output:

```
--------------------
 1.Int
 2.Float
 3.char
--------------------
 Enter Your Choice:1

 Enter Value:4

 Enter Value:5

 Enter Value:2

 Enter Value:7

 Enter Value:1

 Sorted Completed.
------------------------------
 Value is:1
 Value is:2
 Value is:4
 Value is:5
 Value is:7
 Do You Want to continue:y
 Enter Your Choice:2

 Enter Value:12.4

 Enter Value:7.8

 Enter Value:0.1

 Enter Value:2.3

 Enter Value:78.20

 Sorted Completed.
------------------------------
 Value is:0.1
 Value is:2.3
 Value is:7.8
 Value is:12.4
 Value is:78.2
 Do You Want to continue:y

 Enter Your Choice:3
Enter String:users

 Sorted Completed.
------------------------------
 Value is:e
 Value is:r
 Value is:s
 Value is:s
 Value is:u

 Do You Want to continue:n
```

****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****

Q(6):	WAP to create base class Book having int id and char name as data members and respective functionality, show following types of inheritance and display

the details of each kind of books, also calculate the total no of each type of books in proper format.

Simple inheritance with derived class Sales

Hierarchical inheritance with derived classes academics and thrillers

Show use of constructor and destructor in above examples of inheritance.

****

```
#include<iostream>

#include<conio.h>

#include<string>

using namespace std;

ostream & line(ostream &);
```

```
class Book
{
protected:
        int id;
        char *name,*book_type;
        float price;

public:
        Book(int id,char *name,float price,char *book_type)
        {
                cout<<line<<"\n Book Class Constructor is call";
                this->id=id;
                this->name=name;
                this->book_type=book_type;
                this->price=price;
        }
        ~Book(){
                cout<<line<<"\n Book Class Destructor is call";
                delete []name;
                delete []book_type;
        }

};

class Sales : public Book
{
protected:
        int qty;
public:
        Sales(int id,char *name,float price,char *book_type,int qty):Book(id,name,price,book_type)
        {
```

```
        cout<<line<<"\n Sales Class Constructor is call";
        this->id=id;
        this->name=name;
        this->price=price;
        this->book_type=book_type;
        this->qty=qty;
    }
    void display_books()
    {

        cout<<"\n"<<line<<"\n Book Details \n"<<line;
        cout<<"\n Id is:"<<this->id;
        cout<<"\n Name is:"<<this->name;
        cout<<"\n Price is:"<<this->price;
        cout<<"\n Type:"<<this->book_type;
        cout<<"\n Qty is:"<<this->qty;
        cout<<"\n"<<line;
    }
    ~Sales()
    {
        cout<<line<<"\n Sales Class Destructor is call";
        delete []name;
        delete []book_type;
    }

};

class Academics : public Book
{

public:
        Academics(int id,char *name,float price):Book(id ,name,price,"Acadamics")
        {
                cout<<"\n Academics  Calss Constructor is call";
        }

        ~Academics(){
                cout<<"\n Academics Class Destructor is call";
                delete []name;
                delete []book_type;
        }
};

class Thrillers : public Book
{

public:
        Thrillers(int id,char *name,float price):Book(id ,name,price,"Thrillers")
        {
                cout<<line<<"\n Thrillers Class Constructor is call";
        }

        ~Thrillers(){
                cout<<line<<"\n Thrillers  Class Destructor is call";
                delete []name;
                delete []book_type;
        }

};

void main()
{
        Book b1(101,"Tom & Jarry",320,"Story");
        Sales s1(101,"Book 1",450,"Programming",5);
        Sales   s2(102,"Book 5",560,"Opps",6);
        s1.display_books();
        Academics a1(103,"Book 2",400);
        Thrillers t1(104,"Book 3",780);

        //s1.display_books();
        getch();
}

ostream & line(ostream &obj)
{
        cout<<"\n";
        for(int i=0;i<50;i++)
        {
                obj<<"-";
        }
        return obj;
}
```

****
****

Output:

```


--------------------------------------------------
 Book Class Constructor is call
--------------------------------------------------
 Book Class Constructor is call
```

```
-------------------------------------------------
 Sales Class Constructor is call
-------------------------------------------------
 Book Class Constructor is call
-------------------------------------------------
 Sales Class Constructor is call


-------------------------------------------------
 Book Details


-------------------------------------------------
 Id is:101
 Name is:Book 1
 Price is:450
 Type:Programming
 Qty is:5


-------------------------------------------------
-------------------------------------------------
 Book Class Constructor is call
 Academics  Calss Constructor is call
-------------------------------------------------
 Book Class Constructor is call
-------------------------------------------------
 Thrillers Class Constructor is call

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

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(7):   WAP to create student having data members (rollno, name, stream) as base class. Derive class subject with marks of 5 subjects and apply respective

functionality. Calculate final result and display details of each student from derived class. (multilevel inheritance)

****
****

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

class Student
{
	int rno;
	char name[20];
	string stream;

public:
	void get_stud()
	{
		cout<<"\n Enter Rollno:";
		cin>>rno;
		cout<<"\n Enter Name:";
		cin>>name;
		cout<<"\n Enter stream:";
		cin>>stream;
	}

	void display_stud()
	{
		cout<<"\n Rollno is:"<<rno;
		cout<<"\n Name is:"<<name;
		cout<<"\n stream is:"<<stream;
	}
};

class Subject :public Student
{
	public:
	int n,marks[5];



	void get_data()
	{
		get_stud();
		cout<<"\n Enter 5 subject marks:";

		for(int i=0;i<5;i++)
		{
			cout<<"\n Enter marks:";
			cin>>marks[i];
		}
	}
	void display()
	{
		display_stud();
	}

};

class Result : public Subject
{
	Subject sub;
	int sum;
	float res;

public:
	void insert_stud()
	{
		sub.get_data();
	}

	void display_stud()
	{
		sub.display();
	}
	void result()
	{
		sum=0;
		for(int i=0;i<5;i++)
		{
			sum=sum+sub.marks[i];
		}
```

```
        res=sum/5;
        cout<<"\n Result is:"<<res;
    }
};

void main()
{
    int n;
    Result r1[5];

    cout<<"\n Enter the total sutudent:";
    cin>>n;
    for(int i=0;i<n;i++)
    {
        cout<<"\n---------"<<i+1<<"--------";
        r1[i].insert_stud();
        cout<<"\n--------------------------\n";
    }

    for(int i=0;i<n;i++)
    {
        cout<<"\n---------"<<i+1<<"--------";
        r1[i].display_stud();
        r1[i].result();
    }
    getch();
}
```

****
****

Output:

```
 Enter the total sutudent:3

---------1--------
 Enter Rollno:1

 Enter Name:shivangi

 Enter stream:science

 Enter 5 subject marks:
 Enter marks:78

 Enter marks:89

 Enter marks:88

 Enter marks:78

 Enter marks:65

-------------------------

---------2--------
 Enter Rollno:2
 Enter Name:riya

 Enter stream:commerce

 Enter 5 subject marks:
 Enter marks:78

 Enter marks:89

 Enter marks:88

 Enter marks:65

 Enter marks:45

--------------------------

---------3--------
 Enter Rollno:3

 Enter Name:priya

 Enter stream:commerce

 Enter 5 subject marks:
 Enter marks:78

 Enter marks:88

 Enter marks:87
 Enter marks:84


 Enter marks:82


-------------------------


---------1--------
 Rollno is:1
 Name is:shivangi
 stream is:science
 Result is:79
---------2--------
 Rollno is:2
 Name is:riya
 stream is:commerce
 Result is:73
---------3--------
 Rollno is:3
 Name is:priya
 stream is:commerce
 Result is:83
```

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(8): An educational institution wishes to maintain a database of its employees. The database is divided into a number of classes whose hierarchical

relationships are shown in fig-1. The figure also shows the minimum information required for each class. Specify all the classes and define function

to create the database and retrieve individual information as and when required. Write parameterized constructor for each class in the hierarchy.

****
****

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

class Staff
{
public:
        int code;
        string name,designation;

public:
        Staff()
        {
        }
        Staff::Staff(int v_code,string v_name,string v_designation)
        {
                code=v_code;
                name=v_name;
        }
        virtual void printdata(){}

        friend class StaffCollection;

};


class Teacher :virtual public Staff

{

public:

        string sub,pub;


public:

        Teacher(int v_code,string v_name,string v_sub,string
v_pub):Staff(v_code,v_name,"Teacher")

        {

                sub=v_sub;

                pub=v_pub;

        }


        void printdata()

        {

                cout<<"Designation is:"<<this->designation<<endl;

                cout<<"------------------------------"<<endl;

                cout<<"code is:"<<code<<endl;

                cout<<"name is:"<<name<<endl;

                cout<<"Designation is:"<<this->designation<<endl;

                cout<<"Subject is:"<<sub<<endl;

                cout<<"Publication is:"<<pub<<endl;

        }


};
```

```
class Officer : public Staff
{
public:
        char grade;

public:
        Officer(int v_code,string v_name,int v_grade):Staff(v_code,v_name,"Officer")
        {
                grade=v_grade;
        }

        void printdata()
        {
                cout<<"Designation is:"<<this->designation<<endl;
                cout<<"------------------------------"<<endl;
                cout<<"code is:"<<code<<endl;
                cout<<"name is:"<<name<<endl;
                cout<<"Designation is:"<<this->designation<<endl;
                cout<<"grade is:"<<grade<<endl;
        }
};

class Typist :public Staff
{
public:
        int speed;
        string typist_type;
public:

        Typist(int v_code,string v_name,int v_speed):Staff(v_code,v_name,"Typist")
        {
```

```
			speed=v_speed;
		}
		virtual	void printdata()=0;

};

class Casual : public Typist
{
public:
		float daily_wages;

public:

		Casual(int v_code,string v_name,int v_speed,float v_wages):Typist(v_code,v_name,v_speed)
		{
			typist_type="Casual";
			daily_wages=v_wages;
		}
		void printdata()
		{
			cout<<"Designation is:"<<designation<<endl;
			cout<<"------------------------------"<<endl;
			cout<<"code is:"<<code<<endl;
			cout<<"name is:"<<name<<endl;
			cout<<"Designation is:"<<this->designation<<endl;
			cout<<"Speed is:"<<speed<<endl;
			cout<<"Type is:"<<typist_type<<endl;
			cout<<"Daily wages is:"<<daily_wages<<endl;
		}
};
```

```
class Reguler : public Typist
{

public:
        Reguler(int v_code,string v_name,int v_speed):Typist(v_code,v_name,v_speed)
        {
                typist_type="reguler";

        }
        void printdata()
        {
                cout<<"Designation is:"<<this->designation<<endl;
                cout<<"------------------------------"<<endl;
                cout<<"code is:"<<code<<endl;
                cout<<"name is:"<<name<<endl;
                cout<<"Designation is:"<<this->designation<<endl;
                cout<<"Speed is:"<<speed<<endl;
                cout<<"Type is:"<<typist_type<<endl;
        }
};

class StaffCollection
{
        Staff *list[20];
        int count,size;
public:

        StaffCollection::StaffCollection(int n){
        *list = new Staff[n];
        this->count=0;
```

```
this->size=n;

}
int getCount() { return this->count; }

void addStaff(Staff *s) {
        if (this->count >= this->size) {
                cout << "Staff is full!" << endl;
                return;
        }
        list[this->count] = s;
        this->count++;
}

void displaysaff()
{
        for(int i=0;i<this->count;i++)
        {
                list[i]->printdata();
        }
}

Staff *readstaff()
{
        Staff *member=NULL;
        string name,subject,publication;
        int speed,op,code;
        float daily_wages;
        char grade;
        cout<<"\n--------------------";
        cout<<"\n 1.Teacher \n 2.Officer \n 3.Reguler Typist \n 4.Casual Typist";
```

```
cout<<"\n--------------------";
cout<<"\n enter your choice:";
cin>>op;

cout<<"\n Enter staff code and name:";
cin>>code>>name;
switch(op)
{
case 1:
            cout<<"\n Subject is:";
            cin>>subject;
            cout<<"\n Publication is:";
            cin>>publication;
            member = new Teacher(code,name,subject,publication);
            break;
case 2:
            cout<<"\n Enter grade:";
            cin>>grade;
            member = new Officer(code,name,grade);
            break;
case 3:
            cout<<"\n Enter Speed:";
            cin>>speed;
            member = new Reguler(code,name,speed);
            break;

case 4:cout<<"\n Enter Speed:";
            cin>>speed;
            cout<<"\n Enter daily wages:";
            cin>>daily_wages;
            member = new Casual(code,name,speed,daily_wages);
```

```
                    break;
            }
            return member;
        }

};
int main()
{
        int n, op = 1;

        cout << "\n Enter total number of staff members: ";
        cin >> n;

        StaffCollection coll(n);
        Staff *member;
        string code;

        while (op) {
            cout << "\n1.Add Staff Member\n2.Display Staff\n3.Exit \n";
            cout<<"\n Enter choice:";
            cin >> op;
            switch (op) {
            case 1: member = coll.readstaff();
                    coll.addStaff(member);
                    break;

            case 2: coll.displaysaff(); break;
            case 3:exit(0);
            default: cout << "Wrong choice!" << endl;
            }
        }
```

```
            _getch();
            return 0;
}
```

****
****

Output:

```

 Enter total number of staff members: 2

1.Add Staff Member
2.Display Staff

 Enter choice:1

-------------------
 1.Teacher
 2.Officer
 3.Reguler Typist
 4.Casual Typist
-------------------
 enter your choice:2

 Enter staff code and name:101 shivangi

 Enter grade:45

1.Add Staff Member
2.Display Staff
```

```
 Enter choice:Wrong choice!


1.Add Staff Member
2.Display Staff


 Enter choice:1


--------------------
 1.Teacher
 2.Officer
 3.Reguler Typist
 4.Casual Typist
--------------------
 enter your choice:4


 Enter staff code and name:102 riya


 Enter Speed:78


 Enter daily wages:4500


1.Add Staff Member
2.Display Staff


 Enter choice:2
Designation is:
------------------------------
code is:101
name is:shivangi
Designation is:
```

```
grade is:4

Designation is:

------------------------------

code is:102

name is:riya

Designation is:

Speed is:78

Type is:Casual

Daily wages is:4500


1.Add Staff Member

2.Display Staff

3.Exit

 Enter choice:3
```

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(9): Consider a class network of fig 2. The class master derives information from both account and admin classes which in turn derived derive information

from the class person. Define all the four classes and write a program to create, update and display the information contained in master objects.

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


#include<iostream>

#include<conio.h>

#include<string>

using namespace std;



class Person

{

public:

            int code;

            string name;

};



class Admin :virtual public Person

{

protected:

            int exp;

};



class Account :virtual public Person

{

protected:

            int pay;


};
```

```
class Master :public Admin, public Account
{
public:
            int get_code()
            {
                  return code;
            }
            void getdata()
            {
                  cout<<"\n Enter Code:";
                  cin>>code;
                  cout<<"\n Enter Name:";
                  cin>>name;
                  cout<<"\n Enter Experience:";
                  cin>>exp;
                  cout<<"\n Enter Pay:";
                  cin>>pay;
            }
            void update_data()
            {     cout<<"\n Enter Name:";
                  cin>>name;
                  cout<<"\n Enter Experience:";
                  cin>>exp;
                  cout<<"\n Enter Pay:";
                  cin>>pay;
            }
            void display()
            {
                  cout<<"\n-------------------------";
                  cout<<"\n Code is:"<<code;
            cout<<"\n Name is:"<<name;
            cout<<"\n Experience is:"<<exp;
            cout<<"\n Pay is:"<<pay;
            cout<<"\n------------------------";
        }
};

void main()
{
    Master m1[5];
    int i,ch,cnt=-1,v_code,res,index=0,flag=0;
    char cont='n';

    cout<<"\n ------------";
    cout<<"\n 1.Add";
    cout<<"\n 2.Update";
    cout<<"\n 3.display";
    cout<<"\n--------------";
    do
    {
    cout<<"\n Enter Your Choice:";
    cin>>ch;

    switch(ch)
    {
        case 1:
                cnt++;
                m1[cnt].getdata();
                break;


        case 2:
                cout<<"\n Enter code:";
                cin>>v_code;
                for(i=0;i<=cnt;i++)
                {
                    res=m1[i].get_code();
                    if(res==v_code)
                    {
                        index=i;
                        flag=1;
                        break;
                    }
                }
                if(flag==1)
                {
                    m1[index].update_data();
                }
                else
                {
                    cout<<"\n Invalid Code";
                }
                break;
        case 3:
                for(i=0;i<=cnt;i++)
                {
                    m1[i].display();
                }
                break;
    }

    cout<<"\n Do You Want to continue:";
    cin>>cont;
```

```
        }while(cont=='y'|| cont=='Y');

}

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

Output:

```
 ------------
 1.Add
 2.Update
 3.display
--------------
 Enter Your Choice:1

 Enter Code:101

 Enter Name:raj

 Enter Experience:4

 Enter Pay:7800

 Do You Want to continue:y

 Enter Your Choice:1

 Enter Code:102
 Enter Name:jinal


 Enter Experience:8


 Enter Pay:5680


 Do You Want to continue:y


 Enter Your Choice:3


-------------------------
 Code is:101
 Name is:raj
 Experience is:4
 Pay is:7800
-------------------------
-------------------------
 Code is:102
 Name is:jinal
 Experience is:8
 Pay is:5680
-------------------------
 Do You Want to continue:y


 Enter Your Choice:2


 Enter code:102


 Enter Name:jinal


 Enter Experience:2
 Enter Pay:5600


 Do You Want to continue:y


 Enter Your Choice:3


------------------------
 Code is:101
 Name is:raj
 Experience is:4
 Pay is:7800
------------------------
------------------------
 Code is:102
 Name is:jinal
 Experience is:2
 Pay is:5600
------------------------
 Do You Want to continue:n


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


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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(10):   Create a class student from which the classes test and sports are derived. The class student has the name and rollno of the student. The class test has

the marks of the internal test and the sports class has the marks of the sports test. The class student contains a virtual function display() which are

implemented in the classes test and sports. Write a program which will take relative information and display it using pointer of the base class.

****
****

```
#include<iostream>

#include<conio.h>

#include<string>

using namespace std;


class Student

{

protected:

        int rno;

        string name;

        public :

                void add()

                {

                        cout<<"\n Enter rollno:";

                        cin>>rno;

                        cout<<"\n Enter name:";

                        cin>>name;

                }

                virtual void display()

                {
```

```
			cout<<"\n rollno is:"<<rno;
			cout<<"\n name is:"<<name;
		}
};

class Test : public Student
{
protected:
	int internal_marks[5];
public:
	void add_marks()
	{
		for(int i=0;i<3;i++)
		{
		cout<<"\n Enter Internal marks:";
		cin>>internal_marks[i];
		}
	}
	void display()
	{
		for(int i=0;i<3;i++)
		{
			cout<<"\n Internal marks is:"<<internal_marks[i];
		}
	}
};

class Sports : public Student
{
protected:
	int sports_mark;
```

```
public:
        void add_sports_marks()
        {
                cout<<"\n Enter sports marks:";
                cin>>sports_mark;

        }
        void display()
        {
                cout<<"\n sports marks is:"<<sports_mark;

        }

};



void main()
{
        Student *s,s1;
        Test t1;
        Sports sp1;

        s=&s1;
        s->add();
        t1.add_marks();
        sp1.add_sports_marks();
        s->display();



        s=&t1;
        s->display();
```

```
            s=&sp1;
            s->display();



            getch();
}
```

****
****

Output:

```
 Enter rollno:1


 Enter name:priya


 Enter Internal marks:45


 Enter Internal marks:89


 Enter Internal marks:78


 Enter sports marks:56


 rollno is:1
 name is:priya
 Internal marks is:45
 Internal marks is:89
 Internal marks is:78
```

sports marks is:56

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

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(11):   Write a Program to perform following operation on text file :

11.1) write content in a text file

11.2) read content from file

11.3) count no of word and no of lines in a file

11.4) copy contents of one file to another file

****
****

```
#include<iostream>

#include<conio.h>

#include<string>

#include<fstream>


using namespace std;


class File
```

```
{
    char inputline[80],outputline[80];
public:
    void write_file(string file)
    {
        ofstream Entryfile(file);
    //  cout<<"Input:"<<endl;
        while(true)
        {
            cin.getline(inputline,80);
            if(!strcmp(inputline, "End"))
                break;
            Entryfile<<"\n"<<inputline;
        }

        Entryfile.close();
    }

    void read_file(string file)
    {
    //  cout<<"Output"<<endl;
        ifstream Displayfile(file);
        while(!Displayfile.eof())
        {
            Displayfile.getline(outputline,80);
            cout<<"\n"<<outputline;
        }

        Displayfile.close();
    }
```

```
void copy_file(string destination,string source)

{

        ofstream Entryfile(destination);

        ifstream Displayfile(source);

        while(!Displayfile.eof())

        {

                Displayfile.getline(outputline,80);

                Entryfile<<outputline<<endl;

        }

        Displayfile.close();

        Entryfile.close();

        cout<<"\n file Copied Successfully.";


}


void count_word_lines(string file)

{

        char ch;

        int lines=0,word=0;


        ifstream Displayfile(file);

        while(!Displayfile.eof())

        {

                Displayfile.unsetf(ios::skipws);

                Displayfile>>ch;


                if(ch ==' ')

                {

                        word++;

                }

                if(ch=='\n')
```

```
            {
                lines++;
                word++;
            }
        }
        cout<<"\n Lines is:"<<lines;
        cout<<"\n Word is:"<<word;
    }

};

void main()
{
    /*
    File fp;

    fp.write_file("data.txt");
    fp.read_file("data.txt");
    fp.copy_file("details.txt","data.txt");
    fp.read_file("details.txt");
    fp.count_word_lines("data.txt");
    getch();
    */

    File fp;
    int ch;
    char choice='n';
    string file,source,destination;



    cout<<"\n -------------------------------------------";
```

```
cout<<"\n 1.write content in a text file ";
cout<<"\n 2.read content from file ";
cout<<"\n 3.count no of word and no of lines in a file";
cout<<"\n 4.copy contents of one file to another file";
cout<<"\n---------------------------------------------";
do
{
cout<<"\n Enter Your Choice:";
cin>>ch;

switch(ch)
{
        case 1:
                        cout<<"\n Enter Filename:";
                        cin>>file;
                        fp.write_file(file);
                        break;
        case 2:
                        cout<<"\n Enter Filename:";
                        cin>>file;
                        fp.read_file("data.txt");
                        break;
        case 3:
                        cout<<"\n Enter Filename:";
                        cin>>file;
                        fp.count_word_lines("data.txt");
                        break;
        case 4:
                        cout<<"\n Enter destination Filename:";
                        cin>>destination;
                        cout<<"\n Enter source Filename:";
```

```
                                        cin>>source;
                                        fp.copy_file("details.txt","data.txt");
                                        break;

            }

            cout<<"\n Do You Want to continue:";
            cin>>choice;
            }while(choice=='y'|| choice=='Y');



getch();
}
```

****
****

Output:

```

 --------------------------------------------
 1.write content in a text file
 2.read content from file
 3.count no of word and no of lines in a file
 4.copy contents of one file to another file
--------------------------------------------
 Enter Your Choice:1

 Enter Filename:data.txt
user is dump
hello world
```

```
End

 Do You Want to continue:y

 Enter Your Choice:2

 Enter Filename:data.txt



user is dump
hello world
 Do You Want to continue:y

 Enter Your Choice:3

 Enter Filename:data.txt

 Lines is:2
 Word is:5
 Do You Want to continue:y

 Enter Your Choice:4

 Enter destination Filename:details.txt

 Enter source Filename:data.txt

 file Copied Successfully.
 Do You Want to continue:y
 Enter Your Choice:2


 Enter Filename:details.txt



user is dump
hello world
 Do You Want to continue:n
```

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(12):   Write a program to create a file student to store name and marks of 5 students and then display them.

****
****

```
#include<iostream>
#include<conio.h>
#include<string>
#include<fstream>

using namespace std;
```

```
class Student
{
        int rno,marks[3];
        string name;
public:
        void getdata()
        {
                cout<<"\n Enter Roll no:";
                cin>>rno;
                cout<<"\n Enter name:";
                cin>>name;

                for(int i=0;i<3;i++)
                {
                        cout<<"\n Enter marks:";
                        cin>>marks[i];
                }
        }
        void display()
        {
                cout<<"\n Roll no is:"<<rno;
                cout<<"\n Name is:"<<name;

                for(int i=0;i<3;i++)
                {
                        cout<<"\n marks of sub "<<i+1<<" is:"<<marks[i];
                }
        }
};
```

```
void main()
{
	Student obj;
	char Continue = 'n';

	//write data into file
	ofstream enterfile;
	enterfile.open("student.txt",ios::out || ios::binary || ios::trunc);
	if(!enterfile.is_open()){
		cout<<"Unable to open a file.";
	}
	else
	{
		cout<<"\n Input Data:";
		do
		{

			obj.getdata();
			enterfile.write((char *)&obj,sizeof(obj));
			if(enterfile.fail())
			{
				cout<<"\n File write failed.";
			}

			cout<<"\n Do you want to continue(y/n):";
			cin>>Continue;
		}while(Continue=='y');
	}
	enterfile.close();

	//display data from file
```

```
        ifstream display("student.txt",ios::in || ios::binary);
        cout<<"\n Output:\n";
        while(!display.eof())
        {
                display.read((char *)&obj,sizeof(obj));
                if(display.fail())
                        break;
                obj.display();



        }
        display.close();
        getch();
}
```

****
****

Output:

```
 Input Data:
 Enter Roll no:1


 Enter name:sivangi


 Enter marks:89


 Enter marks:78
```

```
Enter marks:56

Do yu want to continue:(y/n)y

Enter Roll no:2

Enter name:jinal

Enter marks:45

Enter marks:56

Enter marks:78

Do you want to continue(y/n):n

Output:

Roll no is:1
Name is:sivangi
marks of sub 1 is:89
marks of sub 2 is:78
marks of sub 3 is:56

Roll no is:2
Name is:jinal
marks of sub 1 is:45
marks of sub 2 is:56
marks of sub 3 is:78
```

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(13):   Define a class Result which contains the result of an MCA II written test. It should take list from a file and display on the screen such that at a time only

ten candidates information is printed on the screen.

****
****

```
#include<iostream>

#include<conio.h>

#include<string>

#include<fstream>


using namespace std;


class Result

{

        float per;

        int rno,marks[3],sum;

        string name;

public:

        void getdata()

        {

                sum=0;
```

```
		cout<<"\n Enter Roll no:";
		cin>>rno;
		cout<<"\n Enter name:";
		cin>>name;

		for(int i=0;i<3;i++)
		{
			cout<<"\n Enter marks:";
			cin>>marks[i];
			sum=sum+marks[i];
		}
		per=sum/3;
	}

	void result()
	{
		cout<<"\n"<<rno<<"\t"<<name<<"\t"<<per;
	}
};

void main()
{
	int cnt=0;
	Result obj;
	char Continue = 'n';

	//write data into file
	ofstream enterfile;
	enterfile.open("student.txt",ios::out || ios::binary || ios::trunc);
	if(!enterfile.is_open()){
		cout<<"Unable to open a file.";
```

```
}
else
{
        cout<<"\n Input Data:";
        do
        {

                obj.getdata();
                enterfile.write((char *)&obj,sizeof(obj));
                if(enterfile.fail())
                {
                        cout<<"\n File write failed.";
                }

                cout<<"\n Do you want to continue:(y/n)";
                cin>>Continue;
        }while(Continue=='y');
}
enterfile.close();

//display data from file

ifstream display("student.txt",ios::in || ios::binary);
cout<<"\n Output:\n";
cout<<"\n rno \t name \t per";
while(!display.eof() && cnt<10)
{
        cnt++;
        display.read((char *)&obj,sizeof(obj));
        if(display.fail())
                break;
        obj.result();



    }
    display.close();
    getch();
}
```

****
****

Output:

```
 Input Data:
 Enter Roll no:1

 Enter name:flora

 Enter marks:45

 Enter marks:89

 Enter marks:78

 Do you want to continue:(y/n)y

 Enter Roll no:2

 Enter name:heena
 Enter marks:45

 Enter marks:56

 Enter marks:2

 Do you want to continue:(y/n)
y

 Enter Roll no:3

 Enter name:mira

 Enter marks:47

 Enter marks:89

 Enter marks:66

 Do you want to continue:(y/n)y

 Enter Roll no:4

 Enter name:rajvi

 Enter marks:45

 Enter marks:78

 Enter marks:89
```

```
Do yu want to continue:(y/n)y

Enter Roll no:5

Enter name:meet

Enter marks:45

Enter marks:12

Enter marks:23

Do you want to continue:(y/n)y

Enter Roll no:6

Enter name:jinali

Enter marks:45

Enter marks:55

Enter marks:55

Do you want to continue:(y/n)y

Enter Roll no:7

Enter name:aditi

Enter marks:80
Enter marks:89

Enter marks:87

Do you want to continue:(y/n)y

Enter Roll no:8

Enter name:preet

Enter marks:56

Enter marks:65

Enter marks:54

Do you want to continue:(y/n)y

Enter Roll no:9

Enter name:uttam

Enter marks:78

Enter marks:89

Enter marks:66

Do you want to continue:(y/n)y
```

```
Enter Roll no:10

Enter name:janvi

Enter marks:78

Enter marks:88

Enter marks:87

Do you want to continue:(y/n)y

Enter Roll no:11

Enter name:kiraa

Enter marks:45

Enter marks:56

Enter marks:78

Do you want to continue:(y/n)n

Output:
rno              name    per
1     flora   70
2     heena   34
3     mira    67
4     rajvi   70
5     meet    26
```

```
6      jinali  51
7      aditi   85
8      preet   58
9      uttam   77
10      janvi   84
```

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(14):   Use an Employee Class to write records of employee to a file. Include a menu that will allow the user to select any of the following features

a. Add a new record.

b. Modify an existing record.

c. Retrieve and display an entire record for a given name.

d. Generate a complete list of all names, addresses and telephone numbers.

e. End of the computation.

****
****

```
#include<iostream>

#include<string>

#include<fstream>

using namespace std;

class emp{
        int id;
        string name,address;
public:
        void getdata(){
                cout<<"Enter the emp id: ";
                cin>>id;
                cout<<"Enter the name: ";
                cin.ignore();
                getline(cin,name);
                cout<<"Enter the Address:";
                cin.ignore();
                        getline(cin,address);
        }
        void show(){
                cout<<"id: "<<id<<endl;
                cout<<"Name: "<<name<<endl;
                cout<<"Address: "<<address<<endl;
        }
        int search(int x){
                if(x==id)
                        return 1;
                else
                        return 0;
        }
        void update(string var_change,int choice){
                if(choice==1)
                        name=var_change;
                else if(choice==2)
                        address=var_change;

        }
```

```
};
void main(){
        int choice,flag=0,temp_id,position,endposition,n,choice2;
        char var_continue='y';
        string var_change;
        emp obj;
        fstream file;
        do{
                cout<<"1. Add\n2. Display\n3. Update\n4. Exit\n";
                cin>>choice;
                switch(choice){
                        case 1:
                                file.open("emp.txt",ios::in | ios::out | ios::binary | ios::trunc);
                                do{
                                        obj.getdata();
                                        file.write((char*) &obj,sizeof(obj));
                                        if(file.fail())
                                                        cout<<"File write failed";
                                                cout<<"Do you want continue? (y/n): ";
                                                cin>>var_continue;
                                }while(var_continue!='n');
                                file.close();
                                break;
                        case 2:
                                file.open("emp.txt",ios::in | ios::out | ios::binary);
                                cout<<"1. Display Full List\n2. Display Particular Detail\n";
                                cin>>choice2;
                                switch(choice2){
                                        case 1:
                                                file.seekg(ios::beg);
                                                while(file.read((char*) &obj,sizeof(obj))){
```

```
                        obj.show();
                    }
                    break;
                case 2:
                    file.seekg(0,ios::end);
                    endposition=file.tellg();
                    n=endposition/sizeof(emp);
                    cout<<"Enter emp id: ";
                    cin>>n;
                    position=(n-1)*sizeof(emp);
                    file.seekg(position);
                    file.read((char*) &obj,sizeof(obj));
                    obj.show();
                    break;
                default:
                    cout<<"Enter correct choice\n";
            }
            file.close();
            break;
        case 3:
            file.open("emp.txt",ios::in | ios::out | ios::binary);
            file.seekg(0,ios::end);
            endposition=file.tellg();
            n=endposition/sizeof(emp);
            cout<<"Enter emp id: ";
            cin>>n;
            position=(n-1)*sizeof(emp);
            file.seekg(position);
            file.seekp(0);
            file.seekp(position);
            cout<<"Which Field you want to change\n1. Name\n2. Address\n";
```

```
                cin>>choice2;

                switch(choice2){

                    case 1:

                        cout<<"Enter name: ";

                        cin>>var_change;

                        obj.update(var_change,choice2);

                        file.write((char*) &obj,sizeof(obj));

                        break;

                    case 2:

                        cout<<"Enter Address: ";

                        cin>>var_change;

                        obj.update(var_change,choice2);

                        file.write((char*) &obj,sizeof(obj));

                        break;

                    default:

                        cout<<"Enter correct choice\n";

                }


                file.seekg(0);

                file.close();

                break;

            case 4:

                exit(0);

            default:

                cout<<"Enter correct choice\n";

        }

    }while(1);

}
```

****
****

Output:

```
1. Add
2. Display
3. Update
4. Exit
1
Enter the emp id: 101
Enter the name: shivangi
Enter the Address:surat
Do you want continue? (y/n): y
Enter the emp id: 102
Enter the name: heena
Enter the Address:rajkot
Do you want continue? (y/n): n
1. Add
2. Display
3. Update
4. Exit
2
1. Display Full List
2. Display Particular Detail
1
id: 101
Name: shivangi
Address:surat
id: 102
Name: heena
Address:rajkot
```

```
1. Add
2. Display
3. Update
4. Exit
2
1. Display Full List
2. Display Particular Detail
2
Enter emp id: 102
id: 102
Name: heena
Address:rajkot
1. Add
2. Display
3. Update
4. Exit
3
Enter emp id: 101
Which Field you want to change
1. Name
2. Address
1
Enter name: janvi
1. Add
2. Display
3. Update
4. Exit
2
1. Display Full List
2. Display Particular Detail
1
```

```
id: 101

Name: shivangi

Address:surat

id: 102

Name: heena

Address:rajkot

1. Add

2. Display

3. Update

4. Exit

4
```

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(15):   Write a program that stores and displays the records of the customer from a file the following information for account of the customer is to be stored.

Account no, account type, name, old balance, new balance, last payment, date of last payment. Also display the current account status by comparing

current payment and previous balance. Also calculate the current balance by subtracting the current payment from the previous balance.

****
****

```
#include<iostream>

#include<fstream>
```

```
#include<string>
#include<conio.h>
using namespace std;
class customer
{
	int no;
	char ty[20];
	char nm1[50];
	int ob,lp;
	char dt[100];
	public:
	void readscr5()
	{
		static int i;
		cout<<"\n Customer "<<++i;
		cout<<"\n Enter Account Number:";
		cin>>no;

		cout<<"\n Enter Account Type:";
		cin.ignore();
		cin.getline(ty,sizeof(ty));
		cin.ignore();
		cout<<"\n Enter Name:";
		cin.getline(nm1,sizeof(nm1));
		cout<<"\n Enter Old Balance:";
		cin>>ob;
		cout<<"\n Enter Last Payment:";
		cin>>lp;
		cin.ignore();
		cout<<"\n Enter Date of Last Payment(DD MM YYYY):";
		cin.getline(dt,sizeof(dt));
	}
	void writescr5()
	{

	cout<<"\n"<<no<<"\t\t"<<ty<<"\t\t"<<nm1<<"\t"<<ob<<"\t\t"<<lp<<"\t\t"<<dt<<"\t"<<ob-
lp;
	}
	void writef5()
	{
		char ch;
		int n,i;
		ofstream f;
		f.open("customer.txt");

		//write() will put data in file which is taken from object
		if(!f.is_open())
			cout<<"\n File cannot be opened";
		else
		{
			cout<<"\n How many records u want to enter:";
			cin>>n;
			for(i=0;i<n;i++)
			{
				readscr5();
				f.write((char*)this,sizeof(*this));
			}
			//cout<<"\n Records Successfully Inserted in file";
		}
		f.close();
	}
	void bal()
	{
```

```
            ifstream f;
            char ch;
            int Num,Amt,flag=0,Num2,Amt2;
            f.open("customer.txt");
            cout<<"\n Do u want to add Amount in Account?\n1.Yes\n2.No"<<endl;
            cout<<"\n Press 1 or 2:";
            cin>>ch;
            cout<<"\n Enter Account Number:";
            cin>>Num;
            cout<<"\n Enter Amount to add:";
            cin>>Amt;
            while(f.read((char*)this,sizeof(*this))!=NULL)
            {
                if(f.fail())
                    cout << "\nFile write failed";
                if(no==Num)
                {
                    cout<<"\n Current Status of Account Number "<<no<<" is
"<<Amt;
                    cout<<"\n New Balance of Account Number "<<no<<" is
"<<(ob-lp)+Amt;
                    flag=1;
                }

            }
            if(flag==0)
                cout<<"\n Enter valid Account Number";
            f.close();
    }
    void readf5()
    {
        char ch;
        ifstream f;
        f.open("customer.txt");

        //read() will take data from file and will put in object
        if(!f)
            cout<<"\nFile not found";
        else
        {
            f.seekg(0,ios::beg);
            //cout<<"\n"<<f.tellg(); //0
            while(f.read((char*)this,sizeof(*this)) !=NULL)
            {
                    //cout<<"\n"<<f.tellg(); //188
                if(f.fail())
                    cout << "\nFile write failed";
                writescr5();
            }
            bal();
            f.close();
        }

    }
};
void main()
{
    customer r;
    int n,i;
    //write data in file which is taken from object
    cout<<"\n Enter Information to be Inserted in File\n";
    r.writef5();
    cout<<"\n Displaying Records from File";
```

```
	cout<<"\nAccountNumber\tAccountType\tName\toldBalance\tLastpayment\tDate\tCurrent Balance";
	//Read data from file which is going in object and then displaying object information";
	r.readf5();
	cout<<"\n";
}
```

****
****

Output:

****
****

Name   : Pradip S Karmakar.

Roll No : 10

Class   : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(16):   WAP to create namespace having function for total_marks. Show its use in class ‘marks’ of students, display total marks of subjects using namespace.

****
****

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

```
namespace
{
        int total(int marks[],int n)
        {
                int sum=0;

                for(int i=0;i<n;i++)
                {
                        sum=sum+marks[i];
                }
                return sum;
        }
}

class Marks
{
        int rno,marks[5],sum;
public:
        void addmarks()
        {
                cout<<"\n Enter Roll no:";
                cin>>rno;
                for(int i=0;i<5;i++)
                {
                        cout<<"\n Enter Marks:";
                        cin>>marks[i];
                }
        }

        void display()
        {
```

```
                        cout<<"\n Roll no is:"<<rno;
                        for(int i=0;i<5;i++)
                        {
                                cout<<"\n Marks is:"<<marks[i];
                        }
                        sum=total(marks,5);
                        cout<<"\n sum is:"<<sum;
                }

};

void main()
{
        Marks m1;
        m1.addmarks();
        m1.display();
        getch();
}
```

****
****

Output:

```
 Enter Roll no:1

 Enter Marks:45

 Enter Marks:56
 Enter Marks:44


 Enter Marks:52


 Enter Marks:32


 Roll no is:1
 Marks is:45
 Marks is:56
 Marks is:44
 Marks is:52
 Marks is:32
 sum is:229
```

****
****

Name : Pradip S Karmakar.

Roll No : 10

Class : M.C.A - 2

Subject : Object Oriented Concepts and Programming

****
****

Q(17): WAP to define a vector and use it for student class to store and display information about student (Use STL).

****
****

```
#include <iostream>

#include <vector>
```

```
#include <algorithm>

#include<conio.h>

using namespace std;


class Student

{

private:

        int RollNumber;

        float TotalMarks;

public:

        Student(){

        }

        Student(int TempRollNumber, float TempTotalMarks) {

                RollNumber = TempRollNumber;

                TotalMarks = TempTotalMarks;

        }

        void operator = (Student TempStud){

                RollNumber = TempStud.RollNumber;

                TotalMarks = TempStud.TotalMarks;

        }

        bool operator < (Student TempStud) {

                return(TotalMarks < TempStud.TotalMarks);

        }

        friend ostream & operator <<(ostream & TempOut, Student & TempStud);

};


ostream & operator <<(ostream & TempOut, Student & TempStud) {

        TempOut << "The mark of roll number " << TempStud.RollNumber << " is " <<
TempStud.TotalMarks; return TempOut;

}
```

```
void main() {
        vector  <Student>  StudMarks;
        float TempMarks;
        int i = 0;

        for(;;)
        {
                cout << "Enter the mark for roll number " << i + 1 << " Enter \-1 to stop: ";
                cin >> TempMarks;

                if(TempMarks== -1) break;
                StudMarks.push_back(Student(i + 1, TempMarks));
                ++i;
        }

        cout << "The size of StudMarks is " << StudMarks.size()<< endl;

        vector <Student>::iterator index;
        sort(StudMarks.begin(), StudMarks.end());

        for(index = StudMarks.begin(); index < StudMarks.end(); ++index)
                cout << *index << endl;
        //return 0;
        getch();
}
```

****
****

Output:

```
Enter the mark for roll number 1 Enter -1 to stop: 45

Enter the mark for roll number 2 Enter -1 to stop: 55

Enter the mark for roll number 3 Enter -1 to stop: 89

Enter the mark for roll number 4 Enter -1 to stop: -1

The size of StudMarks is 3

The mark of roll number 1 is 45

The mark of roll number 2 is 55

The mark of roll number 3 is 89
```