Assignment 31

1. Define a class Person with instance members name and age. Also define member

functions setName(), setAge(), getName(), getAge(). Now define class Employee by

inheriting Person class. In the Employee class define empid and salary as instance

members. Also define setEmpid, setSalary, getEmpid, getSalary.

#include<iostream>

#include<string.h>

using namespace std;

class person

{

    protected:

        char name[50];

        int age;

    public:

        void setName(char \*n)

        {

            strcpy(name,n);

        }

        void setAge(int a)

        {

            age=a;

        }

        char \* getName()// pointer is used becoused we are returning the base addressof name

        {

            return name;

        }

        int getAge()

        {

            return age;

        }

};

class Emplyoee : public person

{

    int empid;

    float salary;

    public:

        void setEmpid(int n)

        {

            empid=n;

        }

        void setSalary(float s)

        {

            salary=s;

        }

        int getEmpid()

        {

            return empid;

        }

        float getSalary()

        {

            return salary;

        }

        void display()

        {

            cout<<"Name  :  "<<name<<"\n Age  :  "<<age<<"\n ID  :  "<<empid<<"\n Salary  :  "<<salary<<endl;

        }

};

int main()

{

    Emplyoee e1, e2;

    e1.setName("Tushar");

    e1.setAge(23);

    e1.setSalary(500);

    e1.setEmpid(1234);

    e2.setName("Keshav");

    e2.setAge(23);

    e2.setSalary(500);

    e2.setEmpid(1230);

    e1.display();

    e2.display();

    return 0;

}

Output:

Name : Tushar

Age : 23

ID : 1234

Salary : 500

Name : Keshav

Age : 23

ID : 1230

Salary : 500

PS C:\Users\tusha\Documents\coadind\assignment\_31.cpp>

2. Write a C++ program to add two numbers using single inheritance. Accept these two

numbers from the user in base class and display the sum of these two numbers in

derived class.

#include<iostream>

using namespace std;

class add\_data

{

    protected:

        int x,y;

    public:

        void accept()

        {

            cout<<"enter the first value : ";

            cin>>x;

            cout<<"enter the second value : ";

            cin>>y;

        }

};

class addition : public add\_data

{

    int add;

    public:

        void add()

        {

            add=x+y;

        }

        void display()

        {

            cout<<"\n addtion  :  "<<add;

        }

};

int main()

{

    addition a;

    a.accept();

    a.add();

    a.display();

    return 0;

}

3. Write a C++ program to calculate the percentage of a student using multi-level

inheritance. Accept the marks of three subjects in base class. A class will be derived

from the above mentioned class which includes a function to find the total marks

obtained and another class derived from this class which calculates and displays the

percentage of students.

#include<iostream>

using namespace std;

class student

{

    protected:

       int marks[3];

    public:

        void accept()

        {

            cout<<"enter the marks of 3 subject : ";

            for(int i = 0; i < 3; i++)

            {

                cin>>marks[i];

            }

        }

};

class Total\_marks : public student

{

    protected:

        const int total=0;

    public:

        void find\_total()

        {

            for(int  i = 0; i < 3; i++)

            {

                total= total+marks[i];

            }

        }

};

class percentage : public Total\_marks

{

    int per;

    public:

        void per()

        {

            per=total/3.0;

        }

        void display()

        {

            cout<<"\n Percentage :  "<<per<<endl;

        }

};

int main()

{

    percentage p;

    p.accept();

    p.find\_total();

    p.per();

    p.display();

    return 0;

}

4. Write a C++ program to design a base class Person (name, address,

phone\_no). Derive a class Employee (eno, ename) from Person. Derive a

class Manager (designation, department name, basic-salary) from

Employee. Write a menu driven program to:

a. Accept all details of 'n' managers.

b. Display manager having highest salary

input:

#include<iostream>

using namespace std;

class person

{

    protected:

        char name[50];

        char address[100];

        int phone;

};

class emplyoee : public person

{

    public:

        int eno;

        char ename[50];

};

class manager: public emplyoee

{

    public:

    char designation[50];

    char depart\_name[50];

    float basic\_salary;

    public:

        void accept\_details()

        {

            cout<<" Enter the manager details "<<endl;

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

            cout<<"enter the emplyoee no.  :  ";

            cin>>eno;

            cout<<"\n enter the emplyoee name  :  ";

            cin>>ename;

            cout<<"\n enter the address  : ";

            cin>>address;

            cout<<"\n enter designation : ";

            cin>>designation;

            cout<<"\n entr department  :  ";

            cin>>depart\_name;

            cout<<"\n enter basic salary : ";

            cin>>basic\_salary;

        }

};

int main()

{

    int n;

    manager m[10];

    cout<<"enter the manager numbers : ";

    cin>>n;

    for (int  i = 0; i < n; i++)

    {

        m[i].accept\_details();

    }

    int temp=0;

    for (int i = 0; i < n; i++)

    {

        if(m[temp].basic\_salary<m[i].basic\_salary)

            temp=i;

    }

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

    cout<<"maximum salary is  :  "<<m[temp].basic\_salary<<endl;

    cout<<" and employee is  : "<<m[temp].ename;

    return 0;

}

Output;

enter the manager numbers :

5

Enter the manager details

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

enter the emplyoee no. : 12345

enter the emplyoee name : tushar

enter the address : wardha

enter designation : cheif officer

entr department :

enter basic salary : 500

Enter the manager details

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

enter the emplyoee no. : 12456

enter the emplyoee name : pavan

enter the address : sorta

enter designation : officer

entr department : wre

enter basic salary : 600

Enter the manager details

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

enter the emplyoee no. : 12654

enter the emplyoee name : harshad

enter the address : pulgaon

enter designation : officer

entr department : pwd

enter basic salary : 430

Enter the manager details

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

enter the emplyoee no. : 78451

enter the emplyoee name : kunal

enter the address : arvi

enter designation : officer

entr department : engineer

enter basic salary : 460

Enter the manager details

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

enter the emplyoee no. : 95412

enter the emplyoee name : radhe

enter the address : jatni

enter designation : officer

entr department : electrical

enter basic salary : 654

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

maximum salary is : 654

and employee is : radhe

PS C:\Users\tusha\Documents\coadind\assignment\_31.cpp

//5. Write a C++ program to define a base class Item (item-no, name, price).

//erive a class Discounted-Item (discount-percent). A customer purcha

//

//

//

#include<iostream>

using namespace std;

 class Item

 {

    public:

        int item\_no;

        char name[50];

        double price;

 };

 class discount : public Item

 {

    public:

        int discount\_percent;

        double discount\_price;

        void accepted\_details()

        {

            cout<<"enter the item number  :  ";

            cin>>item\_no;

            cout<<"enter the item name   :  ";

            cin>>name;

            cout<<"enter the price        :  ";

            cin>>price;

            cout<<"enter the discount per.:  ";

            cin>>discount\_percent;

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

            discount\_price=price-price\*discount\_percent/100.0;

        }

        void display()

        {

            cout<<"Item no.         : "<<item\_no<<endl;

            cout<<"Item Name        : "<<name<<endl;

            cout<<"price            : "<<price<<endl;

            cout<<"Discount         : "<<discount\_percent<<endl;

            cout<<"discouted price  : "<<discount\_price<<endl;

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

        }

 };

 int main()

 {

    int total\_price=0, total\_dicount=0;

    int n;

    cout<<"enter the number of items you want to purchase :  ";

    cin>>n;

    discount d[n];

    for( int i=0; i<n ; i++)

    {

        d[i].accepted\_details();

    }

    for( int i=0; i<n ; i++)

    {

        d[i].display();

    }

    for( int i=0; i<n ; i++)

    {

        total\_price=total\_price+d[i].price;

    }

    for( int i=0; i<n ; i++)

    {

        total\_dicount=total\_dicount + d[i].discount\_price;

    }

    cout<<"Total price      :  "<<total\_price<<endl;

    cout<<"total dicont     :  "<<(total\_price - total\_dicount)<<endl;

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

    cout<<"price to pay     :  "<<total\_dicount<<endl;

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

    return 0;

 }

5. Write a C++ program to define a base class Item (item-no, name, price).

Derive a class Discounted-Item (discount-percent). A customer purchases

'n' items. Display the item-wise bill and total amount using appropriate

format.

Output:

cd "c:\Users\tusha\Documents\coadind\assignment\_31.cpp\" ; if ($?) { g++ problem05.cpp -o problem05 } ; if ($?) { .\problem05 }

enter the number of items you want to purchase : 2

enter the item number : 1245

enter the item name : jeans

enter the price : 500

enter the discount per.: 10

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

enter the item number : 4521

enter the item name : shirt

enter the price : 600

enter the discount per.: 10

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

Item no. : 1245

Item Name : jeans

price : 500

Discount : 10

discouted price : 450

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

Item no. : 4521

Item Name : shirt

price : 600

Discount : 10

discouted price : 540

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

Total price : 1100

total discount : 110

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

price to pay : 990

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

PS C:\Users\tusha\Documents\coadind\assignment\_31.cpp>

6. Write a C++ program to demonstrate how a common friend function can

be used to exchange the private values of two classes. (Use call by

reference method).

#include<iostream>

using namespace std;

class B;

class A

{

    protected:

        int num1;

    public:

    A()

    {

        num1=10;

    }

    void show()

    {

        cout<<"class A  : "<<num1<<endl;

    }

    friend void swap(A \*num1, B \*num2);

};

class B

{

    protected:

        int num2;

    public:

        B()

        {

            num2=20;

        }

        void show()

        {

            cout<<"clas B   : "<<num2<<endl;

        }

        friend void swap(A \*num1 , B \*num2);

};

void swap(A \*no1, B \*no2)

{

    int temp;

    temp = no1->num1;

    no1->num1=no2->num2;

    no2->num2= temp;

}

int main()

{

    A a;

    B b;

    a.show();

    b.show();

    cout<<"--------------"<<endl;

    swap(&a, &b);

    a.show();

    b.show();

    return 0;

}

Output:

cd "c:\Users\tusha\Documents\coadind\assignment\_31.cpp\" ; if ($?) { g++ problem06.cpp -o problem06 } ; if ($?) { .\problem06 }

class A : 10

clas B : 20

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

class A : 20

clas B : 10

PS C:\Users\tusha\Documents\coadind\assignment\_31.cpp>

7. Write class declarations and member function definitions for a C++ base

class to represent an Employee (emp-code, name).

Derive two classes as Fulltime (daily rate, number of days, salary) and

Parttime (number of working hours, hourly rate, salary).

Write a menu driven program to:

1. Accept the details of ‘n’ employees.

2. Display the details of ‘n’ employees.

3. Search a given Employee by emp-code.

#include<iostream>

using namespace std;

class Emplyoee

{

    public:

        int emp\_no;

        char ename[50];

        void accept\_details()

        {

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

            cout<<"Enter the emplyoee number   : ";

            cin>>emp\_no;

            cout<<"Enter the emplyoee name     : ";

            cin>>ename;

        }

};

class Full\_Time : public Emplyoee

{

    public:

        int daily\_rate;

        int days;

        float fsalary;

        void accept\_f\_salary()

        {

            cout<<"Enter the daily rate      : ";

            cin>> daily\_rate;

            cout<<"Enter the working days    : ";

            cin>>days;

        }

        void calculaute()

        {

            fsalary= daily\_rate\* days;

            cout<<"\n salary    : "<<fsalary<<endl;

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

        }

        void display()

        {

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

            cout<<"\n Employee Number   : "<<emp\_no;

            cout<<"\n Employee Name     : "<<ename;

            cout<<"\n Working days      : "<<days;

            cout<<"\n Daily rate        : "<<daily\_rate;

            cout<<"\n Salary            : "<<fsalary;

            cout<<"\n Status            : FULL-TIME"<<endl;

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

        }

};

class Part\_time : public Emplyoee

{

    public:

        int hours;

        int hourly\_rate;

        float psalary;

        void accept\_p\_salary()

        {

            cout<<"Enter the hourly rate      : ";

            cin>> hourly\_rate;

            cout<<"Enter the working hours    : ";

            cin>>hours;

        }

        void p\_calculate()

        {

            psalary=hourly\_rate\* hours;

            cout<<"Salary    : "<<psalary<<endl;

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

        }

        void display()

        {

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

            cout<<"\n Employee Number   : "<<emp\_no;

            cout<<"\n Employee Name     : "<<ename;

            cout<<"\n Working hours     : "<<hours;

            cout<<"\n Hourly rate       : "<<hourly\_rate;

            cout<<"\n Salary            : "<<psalary;

            cout<<"\n Status            : PART\_TIME"<<endl;

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

        }

};

int main()

{

    int var1=0;

    int var2=0;

    Full\_Time f[5];

    Part\_time p[5];

    int choice;

    do

    {

        cout<<"1. Enter record "<<endl;

        cout<<"2. Display record "<<endl;

        cout<<"3. Search record "<<endl;

        cout<<"4. Quit"<<endl;

        cout<<"select the one option : ";

        cin>>choice;

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

        switch (choice)

        {

        case 1 : cout<<"-----DATA ENTRY------\n";

                cout<<"\n 1. Full Time ";

                cout<<"\n 2. Part Time ";

                cout<<"\n select the option    : ";

                int y;

                cin>>y;

                switch (y)

                {

                case 1: cout<<"You have selected Full Time "<<endl;

                        f[var1].accept\_details();

                        f[var1].accept\_f\_salary();

                        f[var1].calculaute();

                        var1++;

                        break;

                case 2: cout<<"You have selected Part Time "<<endl;

                        p[var2].accept\_details();

                        p[var2].accept\_p\_salary();

                        p[var2].p\_calculate();

                        var2++;

                        break;

                default:

                    break;

                }

                break;

        case 2 :

                for(int i=0;i<var1;i++)

                {

                    f[i].display();

                }

                for (int  i = 0; i < var2; i++)

                {

                    p[i].display();

                }

            break;

        case 3 :

                int z;

                int flag=0;

                cout<<"1. Employee Number"<<endl;

                cout<<"2. Employee Name"<<endl;

                cout<<"Emter the choice     : ";

                cin>>z;

                switch (z)

                {

                case 1 :int num, temp;

                        cout<<"Number  : ";

                        cin>>num;

                        for (int i = 0; i < var1; i++)

                        {

                            if(f[i].emp\_no==num)

                            {

                                flag=1;

                                temp=i;

                                break;

                            }

                        }

                        if(flag==1)

                        {

                            cout<<"\n Employee is present "<<endl;

                            f[temp].display();

                        }

                        else

                        {

                            for(int i=0; i<var2 ; i++)

                            {

                               if(p[i].emp\_no==num)

                               {

                                   flag=1;

                                   temp=i;

                                   break;

                               }

                            }

                           if(flag==1)

                            {

                                cout<<"\n Employee is present "<<endl;

                                p[temp].display();

                            }

                            else

                            {

                                cout<<"Employee is not present "<<endl;

                            }

                        }

                        break;

                case 2 : cout<<"\n Ooops !   sorry ....these feature is not availble right now ."<<endl;

                        cout<<"try with the available option only ....."<<endl;

                        break;

                }

            break;

        }

    }while (choice!=4);

    return 0;

}

Output:

"c:\Users\tusha\Documents\coadind\assignment\_31.cpp\" ; if ($?) { g++ problem07.cpp -o problem07 } ; if ($?) { .\problem07 }

1. Enter record

2. Display record

3. Search record

4. Quit

select the one option : 1

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

-----DATA ENTRY------

1. Full Time

2. Part Time

select the option : 1

You have selected Full Time

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

Enter the emplyoee number : 1

Enter the emplyoee name : tushar

Enter the daily rate : 400

Enter the working days : 30

salary : 12000

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

1. Enter record

2. Display record

3. Search record

4. Quit

select the one option : 1

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

-----DATA ENTRY------

1. Full Time

2. Part Time

select the option : 2

You have selected Part Time

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

Enter the emplyoee number : 2

Enter the emplyoee name : keshav

Enter the hourly rate : 50

Enter the working hours : 400

Salary : 20000

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

1. Enter record

2. Display record

3. Search record

4. Quit

select the one option : 2

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

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

Employee Number : 1

Employee Name : tushar

Working days : 30

Daily rate : 400

Salary : 12000

Status : FULL-TIME

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

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

Employee Number : 2

Employee Name : keshav

Working hours : 400

Hourly rate : 50

Salary : 20000

Status : PART\_TIME

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

1. Enter record

2. Display record

3. Search record

4. Quit

select the one option : 3

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

1. Employee Number

2. Employee Name

Emter the choice : 1

Number : 2

Employee is present

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

Employee Number : 2

Employee Name : keshav

Working hours : 400

Hourly rate : 50

Salary : 20000

Status : PART\_TIME

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

1. Enter record

2. Display record

3. Search record

4. Quit

select the one option : 3

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

1. Employee Number

2. Employee Name

Emter the choice : 2

Ooops ! sorry ....these feature is not availble right now .

try with the available option only .....

1. Enter record

2. Display record

3. Search record

4. Quit

select the one option : 1

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

-----DATA ENTRY------

1. Full Time

2. Part Time

select the option : 3

1. Enter record

2. Display record

3. Search record

4. Quit

select the one option : 4

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

PS C:\Users\tusha\Documents\coadind\assignment\_31.cpp>

8 - In a bank, different customers have savings account. Some customers may

have taken a loan from the bank. So bank always maintain information about

bank depositors and borrowers.

Design a Base class Customer (name, phone-number). Derive a class

Depositor(accno, balance) from Customer.

Again, derive a class Borrower (loan-no, loan-amt) from Depositor.

Write necessary member functions to read and display the details of ‘n’

customers.

#include<iostream>

using namespace std;

class Custmer

{

    public:

        char name[50];

        long long phon\_no;

};

class depositer : public Custmer

{

    public:

        long Acc\_no;

        float balance;

};

class borrower : public depositer

{

    public:

        int loan\_no;

        int loan\_amount;

        void accept\_details()

        {

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

            cout<<"Enter the Name    : ";

            cin>>name;

            cout<<"Enter phone no.   : ";

            cin>>phon\_no;

            cout<<"Enter account no. : ";

            cin>>Acc\_no;

            cout<<"Enter balance     : ";

            cin>>balance;

            cout<<"enter loan no.    : ";

            cin>>loan\_no;

            cout<<"enter loan amount : ";

            cin>>loan\_amount;

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

        }

        void display\_details()

        {

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

            cout<<"Customer Name        : "<<name<<endl;

            cout<<"Customer phone No.   : "<<phon\_no<<endl;

            cout<<"Customer Account No. : "<<Acc\_no<<endl;

            cout<<"Customer balance    : "<<balance<<endl;

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

        }

        void display\_loan()

        {

            cout<<"customer loan details "<<endl;

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

            cout<<"loan Number      : "<<loan\_no<<endl;

            cout<<"loan amount      : "<<loan\_amount<<endl;

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

        }

};

int main()

{

    int n;

    borrower b[5];

    cout<<"enter the number of customer : ";

    cin>>n;

    for (int i = 0; i < n; i++)

    {

        b[i].accept\_details();

    }

    for (int i = 0; i < n; i++)

    {

        b[i].display\_details();

        b[i].display\_loan();

    }

    return 0;

}

Output:

"c:\Users\tusha\Documents\coadind\assignment\_31.cpp\" ; if ($?) { g++ problem08.cpp -o problem08 } ; if ($?) { .\problem08 }

enter the number of customer : 2

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

Enter the Name : tushar

Enter phone no. : 7798018146

Enter account no. : 123

Enter balance : 5000

enter loan no. : 12

enter loan amount : 5000

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

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

Enter the Name : keshav

Enter phone no. : 9865321452

Enter account no. : 534

Enter balance : 985123

enter loan no. : 365

enter loan amount : 6000

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

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

Customer Name : tushar

Customer phone No. : 7798018146

Customer Account No. : 123

Customer balance : 5000

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

customer loan details

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

loan Number : 12

loan amount : 5000

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

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

Customer Name : keshav

Customer phone No. : 9865321452

Customer Account No. : 534

Customer balance : 985123

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

customer loan details

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

loan Number : 365

loan amount : 6000

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

PS C:\Users\tusha\Documents\coadind\assignment\_31.cpp>