**INHERITANCE**

1. A publisher company markets both books and CDs. Create a class publication that stores the name (string) and price (float) of books and CDs. From this class derive two classes book which adds a page counts (type int), and CD which adds bytes (type int). Each of these classes should have a function getdata() to get data from the user and a function putdata() to display its data. Write a main() function to test the classes book and CD by creating instances of them, asking the user to input their data using the function getdata(), and then displaying the data with the function putdata().

## INPUT:

## #include<iostream.h>

## #include<string.h>

## #include<conio.h>

## class Publication

## {

## private:

## char name[50];

## float price;

## public:

## void getData()

## {

## cout<<“\nEnter Name: ”;

## gets(name);

## cout<<“Enter Price: $”;

## cin>>price;

## }

## void putData()

## {

## cout<<“\nName: ”;

## puts(name);

## cout<<“\nPrice: ”<<price;

## }

## };

## class Book : public Publication

## {

## private:

## int pages;

## public:

## void getData()

## {

## Publication::getData();

## cout<<“Enter Pages: ”;

## cin>>pages;

## }

## void putData()

## {

## Publication::putData();

## cout<<“\nPages: ”<<pages;

## }

## };

## class CD : public Publication

## {

## private:

## float bytes;

## public:

## void getData()

## {

## Publication::getData();

## cout<<“Enter the bytes: ”;

## cin>>bytes;

## }

## void putData()

## {

## Publication::putData();

## cout<<“\nBytes: ”<<bytes;

## }

## };

## void main()

## {

## clrscr();

## Book b; CD t;

## b.getData();

## t.getData();

## b.putData();

## cout<<endl;

## t.putData();

## cout<<endl;

## getche();

## }

## OUTPUT:

## 

## Create a base class building that stores the number of floors a building has, the number of rooms and its total square footage. Create a derived class called house that inherits building and stores the number of bedrooms and number of bathrooms. Next, create a derived class called office that inherits building and also stores the number of fire extinguishers and the number of telephone.

## INPUT:

## #include<stdio.h>

## #include<iostream.h>

## #include<conio.h>

## class building

## {

## int nfloors;

## int nrooms;

## int foot,total;

## public:

## void get()

## {

## cout <<"enter the number of floors";

## cin>>nfloors;

## cout<<"enter the number of rooms:";

## cin>>nrooms;

## cout<<"enter the feet:";

## cin>>foot;

## }

## void cal()

## {

## int t;

## t=foot/12;

## total=(nfloors\*nrooms )/t;

## }

## void disp()

## {

## cout<<"the number of floors and rooms and the total footage:";

## cout <<nfloors<<nrooms<<total;

## }

## };

## class house: public building

## {

## int nbedrooms;

## int nbathrooms;

## public:

## 

## void get()

## {

## cout<<"enter the number of bedrooms:";

## cin>>nbedrooms;

## cout<<"enter the number of bathrooms:";

## cin>>nbathrooms;

## }

## void put()

## {

## cout<<"the number of bedrooms and bathrooms:";

## cout<<nbedrooms<<nbathrooms;

## }

## };

## class office: public house

## {

## int nfireext;

## int ntele;

## public:

## void get()

## {

## cout<<"enter the number of fire extinguishers:";

## cin>>nfireext;

## cout<<"enter the number of telephones:";

## cin>>ntele;

## }

## void put()

## {

## cout<<"the number of fire extinguishers and telephones:";

## cout<< nfireext << ntele ;

## }

## };

## void main()

## {

## clrscr();

## office off;

## off.get();

## off.put();

## getche();

## }

## OUTPUT:

## 

## Write a program that reads the data of a student and computes its grade using single inheritance.

## INPUT:

## #include <iostream.h>

## #include<conio.h>

## class basic

## {

## private:

## char name[30];

## int age;

## char gender;

## public:

## void getBasicInfo()

## {

## cout << "Enter student's basic information:" << endl;

## cout << "Name: ";

## cin >> name;

## cout << "Age: ";

## cin >> age;

## cout << "Gender: ";

## cin >> gender;

## }

## void putBasicInfo()

## {

## cout << "Name: " << name << ",Age: " << age << ",Gender: " << gender << endl;

## }

## };

## class result : public basic

## {

## private:

## int totalM;

## float perc;

## char grade;

## public:

## void getResultInfo();

## void putResultInfo()

## {

## cout << "Total Marks: " << totalM << ",Percentage: " << perc << ",Grade: " << grade << endl;

## }

## };

## void result::getResultInfo()

## {

## cout << "Enter student's result information:" << endl;

## cout << "Total Marks?: "; cin >> totalM;

## perc= (float)((totalM)/5);

## cout << "Grade?: ";cin >> grade;

## }

## void main()

## {

## clrscr();

## result std;

## std.getBasicInfo();

## std.getResultInfo();

## std.putBasicInfo();

## std.putResultInfo();

## getche();

## }

## OUTPUT:

## 

## 4. A college maintains a list of its students graduating every year. At the end of the year, the college produces a report that lists the following:

## Number of working Graduates:

## Number of non-working graduates:

## Name:

## Age:

## Subject:

## Average Marks:

## X % of the graduates this year are non-working and n% are first divisioners.

## Write a C++ program for it that uses the following inheritance path:

## Person ----->    Student -----🡪           Graduate

## (name, age)   (rollno, avg marks)    (student, employed)

## INPUT:

## #include<iostream.h>

## #include<conio.h>

## #include<stdio.h>

## class Person

## {

## protected:

## char name[20];

## int age;

## public:

## void get()

## {

## cout<<"\n Enter Name: ";

## gets(name);

## cout<<" Enter Age: ";

## cin>>age;

## }

## void put()

## {

## cout<<"\n Name: ";

## puts(name);

## cout<<" Age: "<<age;

## }

## };

## class Student: public Person

## {

## protected:

## int rollno;

## float avgmarks;

## public:

## 

## float avg()

## {

## return avgmarks;

## }

## void get()

## {

## Person::get();

## cout<<" Enter roll no.: ";

## cin>>rollno;

## cout<<" Enter average marks: ";

## cin>>avgmarks;

## }

## void put()

## {

## Person::put();

## cout<<"\n Average Marks: "<<avgmarks;

## }

## };

## class Graduate : public Student

## {

## public:

## char subject[20];

## char employed;

## void get()

## {

## Student::get();

## cout<<" Enter Subject: ";

## gets(subject);

## cout<<" Employed? (Y/N) ";

## cin>>employed;

## }

## void put()

## {

## Student::put();

## cout<<"\n Subject: ";

## puts(subject);

## }

## };

## void main()

## {

## clrscr();

## Graduate g[100];

## int n, t, x, d = 0, work = 0, nwork = 0;

## cout<<"\n Enter the no. of students: ";

## cin>>t;

## for(int i = 0; i < t; i++)

## {

## g[i].get();

## if(g[i].employed == 'y' || g[i].employed == 'Y')

## work++;

## else

## nwork++;

## if(g[i].avg() >= 60)

## d++;

## }

## clrscr();

## cout<<"\n Number of working graduates: "<<work;

## cout<<"\n Number of non-working graduates: "<<nwork;

## n = (d \* 100)/t;

## x = (nwork \* 100)/t;

## cout<<"\n Percentage of First Divisioners: "<<n<<"%";

## cout<<"\n Percentage of non-working graduates: "<<x<<"%";

## for(int j = 0; j < t; j++)

## {

## g[j].put();

## }

## getche();

## }

## OUTPUT:

## 

## 5. Write a C++ program to read and display information about employee and managers. Employee is a class that contains employee number, name, address and department. Manager class contains all information of the employee class and a list of employees working under a manager.

## INPUT:

## #include<iostream.h>

## #include<conio.h>

## #include<stdio.h>

## class Employee

## {

## int no;

## char name[20];

## char address[50];

## int department;

## public:

## int dept()

## {

## return department;

## }

## void get()

## {

## cout<<"\n Enter Employee No.: ";

## cin>>no;

## cout<<" Enter Employee Name: ";

## gets(name);

## cout<<" Enter Address: ";

## gets(address);

## cout<<" Enter Department (1/2/3) : ";

## cin>>department;

## }

## void put()

## {

## cout<<"\n Employee NO.: "<<no;

## cout<<"\n Name: ";

## puts(name);

## cout<<"\n Address: ";

## puts(address);

## cout<<"\n Department: "<<department;

## }

## };

## class Manager: public Employee

## {

## public:

## char mname[20];

## int employees;

## int depthead;

## void get()

## {

## cout<<"\n Enter Name: ";

## gets(mname);

## cout<<" Which Department do you lead? (1/2/3) ";

## cin>>depthead;

## }

## void put()

## {

## cout<<"\n\n Manager Name: ";

## puts(mname);

## cout<<" Head of Department "<<depthead;

## }

## };

## void main()

## {

## clrscr();

## int num, n, m1 = 0, m2 = 0, m3 = 0;

## Employee E[100];

## Manager M[100];

## cout<<"\n Enter the no. of Employees: ";

## cin>>num;

## for(int i = 0; i < num; i++)

## {

## E[i].get();

## if(E[i].dept() == 1)

## m1++;

## else if(E[i].dept() == 2)

## m2++;

## else if(E[i].dept() == 3)

## m3++;

## }

## cout<<"\n Enter the no. of Managers: ";

## cin>>n;

## for(int j = 0; j < n; j++)

## {

## M[j].get();

## if(M[j].depthead == 1)

## M[j].employees = m1;

## else if(M[j].depthead == 2)

## M[j].employees = m2;

## else if(M[j].depthead == 3)

## M[j].employees = m3;

## }

## clrscr();

## for(int k = 0; k < n; k++)

## {

## M[k].put();

## cout<<"\n No. of employees under manager = "<<M[k].employees;

## }

## getche();

## }

## OUTPUT:

## 