**MCQ SECTION**

Q1. C

Q2. C

Q3. B

Q4. C

**CODING SECTION**

***PROBLEM STATEMENT-1(5-MARKS)***

**#include <iostream>**

**using namespace std;**

**class bill**

**{**

protected:

int item\_price,qty,bill;

public:

void get\_bill();

};

class cash:protected bill

{

public:

int a,b,c,d;

void get\_cash()

{

cin>>item\_price>>qty>>a>>b>>c>>d;

}

void display()

{

bill=item\_price\*qty;

//cout<<"Bill: "<<bill;

int t;

t=2000\*a+100\*b+50\*c+10\*d;

//cout<<"\nCash: "<<t;

if(t>=bill)

cout<<”Clear";

else

cout<<"Need to pay: "<<bill-t;

}

};

**int main()**

**{**

**cash d;**

**d.get\_cash();**

**d.display();**

**return 0;**

**}**

***PROBLEM STATEMENT-2(10-MARKS)***

**#include<iostream>**

**#include<iomanip>**

**using namespace std;**

**class university**

**{**

**protected:**

**char uni\_name[20];**

**public:**

**virtual void clerk()=0;**

**virtual void read()=0;**

**};**

**class emp:public university**

**{**

protected:

char name[100];

int age;

int sal;

public:

void read()

{

cin.getline(uni\_name,20);

cin.getline(name,100);

cin>>age;

cin>>sal;

}

};

class cse:public emp

{

public:

void clerk()

{

cout<<"University: "<<uni\_name;

cout<<"\nDept: CSE";

cout<<"\nName: "<<name;

cout<<"\nAge: "<<age;

cout<<"\nSalary: "<<sal<<endl;

}

};

class ece:public emp

{

public:

void clerk()

{

cout<<"University: "<<uni\_name;

cout<<"\nDept: ECE";

cout<<"\nName: "<<name;

cout<<"\nAge: "<<age;

cout<<"\nSalary: "<<sal<<endl;

}

};

class me:public emp

{

public:

void clerk()

{

cout<<"University: "<<uni\_name;

cout<<"\nDept: ME";

cout<<"\nName: "<<name;

cout<<"\nAge: "<<age;

cout<<"\nSalary: "<<sal<<endl;

}

};

**int main()**

**{**

**university \*ptr;**

**int ch;**

**cin>>ch;**

**cin.ignore();**

**switch(ch)**

**{**

**case 1: {**

**cse c;**

**ptr=&c;**

**ptr->read();**

**ptr->clerk();**

**break;**

**}**

**case 2:{**

**ece e;**

**ptr=&e;**

**ptr->read();**

**ptr->clerk();**

**break;**

**}**

**case 3: {**

**me m;**

**ptr=&m;**

**ptr->read();**

**ptr->clerk();**

**break;**

**}**

**default:**

**cout<<"INVALID CHOICE";**

**}**

**return 0;**

**}**