ASSIGNMENT - 3

Name :- Mamta Kumari

Roll NO:- 21053288

Section :- CSE 21

Subject :- OOP Theory

Name:- Mamta Kumari

(1)

```
#include<iostream>
using namespace std;
class vehicle
{
  public:
   int mileage;
   int price;
   void getdata()
  {
   cout << "Enter the mileage of vehicle" << endl;
   cin >> mileage
   cout << "Enter the price of vehicle :" << endl;
   cin >> price;
  }
 void showdata()
 {
   cout << "MILEAGE =" << mileage << endl;
   cout " PRICE = " << Price << endl;
    }
 };
class car : public virtual vehicle
{
 public:
 int owner_cost;
 int warr;
 char * fuel_type;
 void getdata_car()
{
```

```
    fuel_type = new char[50];
    cout<<"Enter the ownership cost of car :"<<endl;
    cin>>owner_cost;
    cout<<"Enter the warranty of the car :"<<endl;
    cin>>warr;
    cout<<"Enter the fuel type of the car :"<<endl;
    getch();
    gets(fuel_type);
  }
  void showdata_car()
  {
  cout<<"OWNERSHIP-COST="<<owner_cost<<endl;
  cout<<"WARRENTY="<<warr<<endl;
  cout<<"FUEL-TYPE="<<fuel_type<<endl;
  }
};
class Bike : public virtual vehicle
{
  public:
  int cylind;
  int gears;
  char wtype[5];
  int fuel_size;
  void getdata_bike()
  {
  cout<<"Enter the number of cylinders:"<<endl;
  cin>>cylind;
  cout<<"Enter the number of gears:"<<endl;
  cin>>gears;
  cout<<"Enter the type of wheel in Bike:"<<endl;
  getchar();
```

```
    gets(wtype);
    cout<<"Enter the fuel tank size of the Bike : "<<endl;
    cin>>fuel_size;
  }
  void showdata_bike()
 {
  cout<<" for Bike :- "<<endl;
  cout<<"NUMBER of CYOLINDERS = "<<cylind<<endl;
  cout<<" NumBER of GEARS = "<<gears<<endl;
  cout<<"TYPE of WHEEL = "<<wtype<<endl;
  cout<<"FUEL SIZE"<<fuel_size<<endl;
  }
};
class Audi : PUBLIC car
{
 PUBLIC :
    char modeltype [50];
    void getdata_audi()
 {
  cout<<"Enter model type :"<<endl;
  gets(modeltype);
  cout<<endl;
 }
 void showdata_audi()
{
  cout<<"for Audi"<<endl;
  cout<<"MODEL TYPE ="<<modeltype<<endl;
 }
};
class ford : PUBLIC car
{
 PUBLIC :
  char modeltype [50];
  void getdata_ford()
 {
```

```
    cout << "Enter model type : " << endl;
    gets (modeltype);
    cout << endl;
  }
  void showdata_ford()
  {
   cout << "for ford " << endl;
   cout << "MODEL TYPE =" << model type << endl;
  }
};
class BAJAJ : public Bike
 { public:
     char modeltype[50];
     void getdata_bajaj()
   {
    cout << "Enter model type : " << endl;
   getchar();
   gets (modeltype);
   cout << endl;
 }
 void showdata_bajaj()
{
 cout << "for BAJAJ " << endl;
 cout << "MODEL TYPE=" << modeltype << endl;
   }
};
class TVS : public BIKE
{
  public:
  char modeltype[50];
  void getdata_tvs()
 {
   cout << "Enter model type : " << end2;
  getchar();
  gets (modeltype);
   cout << end1;
}
```

```
    void showdata_tvs()
    {
      cout<<"for TVS"<<endl;
      cout<<"MODEL TYPE="<<model.type<<endl;
    }
};
int main()
{
  int option, model;
  cout<<"1. Car\n";
  cout<<"2. Bike\n";
  cout<<"Enter your choice:";
  cin>>option;
  if(option==1)
  {
   cout<<"1. Audi\n";
   cout<<"2. Ford\n";
   cout<<"Enter the model :"<<endl;
   cin>>model;
   if(model==1)
   {
   Audi a1;
   a1.getdata();
   a1.getdata_car();
   a1.getdata_audi();
   a1.showdata();
   a1.showdata_car();
   a1.showdata_audi();
  }
  if(model==2)
  {
    ford a1;
    a1.getdata();
```

```
        a1.getdata_car();
        a1.getdata_ford();
        a1.showdata();
        a1.showdata_car();
        a1.showdata_ford();
      }
    }
    if (option==2)
    {
      cout<<"1.Bajaj\n";
      cout<<"2.TVS\n";
      cout<<"Enter the model"<<endl;
      cin>>model;
      if (model==1)
      {
        Bajaj b1;
        b1.getdata();
        b1.getdata_bike();
        b1.getdata_bajaj();
        b1.showdata();
        b1.showdata_bike();
        b1.showdata_bajaj();
      }
      if (model==2)
      {
        TVS b1;
        b1.getdata();
        b1.getdata_bike();
        b1.getdata_tvs();
        b1.showdata();
        b1.showdata_bike();
        b1.showdata_tvs();
      }
    }
}
```

2

```
#include<iostream>
using namespace std;
class shape
{
public:
  virtual void calculate()
{
  cout << "Area of your shape";
}
virtual~shape()
{
 cout << "shape destructor call in";
}
};
class Rectangle : public shape
{
public:
 int width, height, area;
 void calculate()
{
 cout << "Enter width of Rectangle:";
 cin >> width;

 cout << "Enter Height of Rectangle:";
 cin >> height;

 area = height * width;
 cout << "Area of Rectangle:" << area << "in";
}
virtual ~ Rectangle()
{
  cout << "Rectangle destructor call in";
}
};
```

```
class square : public shape
{
public:
int side, area;
void calculate()
{
cout << "Enter one side your of square :";
cin >> side;
area = side * side;
cout << "Area of square :" << area << "\n";
}
virtual ~square()
{
cout << "square destructor call \n";
}
};
int main()
{
shape *s;
Rectangle r;
s = &r;
s -> calculate();
square sq;
s = &sq;
s -> calculate();
return 0;
}
```

(3)

```
#include <iostream>
Using namespace std;
Class Student
{
 Public :
 Char name [50];
 Char branch (20);
 int roll_no;
 virtual void getdata()
{
 cout << " Enter name : " << endl;
 gets (name);
 cout << " Enter branch : " << endl;
 gets (branch);
 cout << " Enter the Roll_NO; " << endl;
  cin >> roll_no;
 }
 virtual void Showdata()
 {
  cout << " In Details of the Student are : " << endl << endl;
  cout << " Name of Student : " << name << endl;
  cout << " Branch of Student : " << branch << endl;
  cout << " Roll no of the Student : " << roll_no << endl;
 }
};
 Class Academic : Public Student
{
 Public :
   int mark_in_maths;
   int marks_in_opps;
   int marks_in_dsa;
   int marks_in_de;
```

```
    void getdata()
    {
    cout<<"\n Enter marks in maths :"<<endl;
    cin >> marks_in_maths;
    cout<<" Enter marks in oop:"<<endl;
    cin >> marks_in_oops;
    cout<<" Enter marks in DSA :"<<endl;
    cin >> marks_in_dsa;
    cout<<"Enter marks in DE :"<<endl;
    cin >> marks_in_de;
    }
    void showdata()
{
int tot, per;
tot = marks_in_de + marks_in_dsa + marks_in_maths +
        marks_in_oops ;
per = tot /4;
cout<<"\n marks in maths :"<<marks_in_maths<<endl;
per = tot /4;
cout<<"\n marks in oop :"<<marks_in_oops<<endl;
cout<<"\n marks in DSA :"<<marks_in_dsa<<endl;
cout<<"\n in DE :"<<marks_in_de<<endl;
cout<<" Total marks :"<<tot<<endl;
cout<<" percentage :"<<per<<endl;
    }
};
int main()
{
  Student *s = new Student;
  Academics a;
  s->getdata();
  s->showdata();
  s= &a;
  s->getdata();
  s->showdata();
```

(5)

```
#include<iostream>
using namespace std;
template <class T>
class vector
{
  T v[20];
  int size;
public:
  void create();
  void modify();
  void mult();
  void display();
};
template<class T>
void vector<T>::create()
{
 int i;
 T value;
 char ans;
 size = 0;
 do
 {
 cout<<"\n Enter the index & value:";
 cin>>i>>value;
 v[i]=value;
 size++;
 cout<<"\n Do you want more element?:";
 cin>>ans;
 } while(ans=='y'||ans=='Y');
}
template<class T>
void vector<T>::modify()
 {
```

```
    int key;
    T newval;
    cout << "\n Enter index for modification:";
    cin >> key;
    cout << "\n Enter new value :";
    cin >> newval;
    V[key] = newval;
  }
template < class T >
void vector<T>:: mult()
{
  int i;
  in scalarval;
  cout << "\n Enter scalar value for multiplication";
  cin >> scalarval;
  for(i=0; i<size; i++)
    v[i] = v[i] * scalarval;
  }
template < class T >
void vector<T>:: display()
{
  int i;
cout << "\n size of vector is :" << size;
cout << "\n Elements in vector are:";
cout << "(";
for (i=0; i<size; i++)
{
cout << v[i] << " ";
}
 cout << ")";
}
int main()
{
```

```
int ch;
vector<int> obj;
cout<<"\n Program for template class";
do
{
 cout<<"\n MAIN MENU";
 cout<<"\n 1. Create";
 cout<<"\n 2. Display";
 cout<<"\n 3. Mult.";
 cout<<"\n 4. Modify";
 cout<<"\n 0. Exit";
 cout<<"\n Enter your choice:";
 cin>>ch;

switch(ch)
{
case 1;
 obj.create();
 break;
Class 2:
obj.display();
break;

Case 3:
  Obj.mult();
  break;

Case 4;
 obj.modify();
 break;

Case 0;
cout<<"\n Exit\n";
 break;
```

```
        default:
          cout << "\n Invalid choice";
          break;
      }
    } while (ch != 0);
    return 0;
}
```

(6)

```
#include<iostream>
using namespace std;
void function()
{
 try
  {
  throw "hello";
 }
catch(const char *)
{
cout<<"In caught exception inside function\n";
throw;
}
}
int main()
{
cout<<"main start";
try
{
  function();
}
catch(const char *)
{
 cout<<"In caught exception inside main\n";
}
cout<<"main end";
return 0;
}
```

7

```
#include <iostream>
using namespace std;
template <typename T> T insert_arr(T a[], int n)
{
cout << "Enter the elements in array" << endl;
for(int i=0; i<n; i++)
{
cin >> a[i];
}
return 0;
}
template <typename T> T search_arr(T a[], T p, int n)
{
int flag=0;
for(int i=0; i<n; i++)
{
if(a[i]==p)
{
flag++;
break;
}else
{
flag=0;
}
}
if(flag!=0)
{
cout << "element is found." << endl;
}else
{
```

```
cout << "Element is not found :" << endl;
}
return 0;
}
int main()
{
int a[100];
int n,p;
cout << "Enter the number of elements:" << endl;
cin >> n;
Insert_arr<int>(a,n);
cout << "Enter the element you want to search" << endl;
cin >> p;
search_arr<int>(a,p,n);
}
```

8)

```
#include<iostream>
using namespace std;
class comparestring
{
public:
char str[25];
comparestring(char str1[])
{
 strcpy(this->str, str1);
}

int operator==(comparestring s2)
{
 if(strcmp(str, s2.str)==0)
 return 1;
else
 return 0;
}

int operator<=(comparestring s3)
{
 if(strlen(str)<=strlen(s3.str))
 return 1;
else
 return 0;
}
 int operator>=(comparestring s3)
{
 if(strlen(str)>=strlen(s3.str))
 return 1;
 else
 return 0;
}
```

```
void compare(CompareString S1, CompareString S2)
{
  if(S1==S2)
  cout<<S1.str<<" is equal to"
      <<S2.str<<endl;
  else
  {
   cout<<S1.str<<" is not equal to"
       <<S2.str<<endl;
   if(S1>=S2)
    cout<<S1.str<<" is greater than"
        <<S2.str<<endl;
   else
    cout<<S2.str<<" is greater than"
        <<S1.str<<endl;
  }
}
  void testCase1()
{
 char str1[]="Hello";
 char str2[]="world";
 CompareString S1(str1);
 CompareString S2(str2);

cout<<"Comparing \""<<S1.str<<"\" and \""
  <<S2.str<<"\""<<endl;
 compare(S1,S2);
}

void testCase2()
{
  char str1[]="Hello";
  char str2[]="Hello";
```

```
    CompareString s1(Str1);
    CompareString s2(Str2);

    cout<<"\n\n Comparing \""<<s1.Str<<"\" and \""
        <<s2.Str<<"\""<<endl;

    Compare(s1, s2);
}

int main()
{
    testCase1();
    testCase2();

    return 0;
}
```