National University of Computer & Emerging Sciences, Karachi

Spring-2025 School of Computing

Re-Quiz

9th May 2025

Course Code: CS-1004	Course Name: Object Oriented Programming
Instructor Name / Names: Ms. Atiya Jokhio	
Section-	**Student-ID:**

Time Allowed: 30 minutes. **Total Points**: 10

Question:1 Predict the output **[4 Points]**

```
#include<iostream>
using namespace std;
 class A{
 public:
 int f(){
 return 1;}
 virtual int g(){
 return 2;}
 };
 class B: public A{
 public:
 int f(){
 return 3;}
 virtual int g(){
 return 4;}
 };
 class C: public A{
 public:
 virtual int g(){
 return 5;}
 };
 int main(){
 A *pa;
 A a;
 B b;
 C c;
 pa=&a;
```

```
int main(){
 A *pa;
 A a;
 C c;
 pa=&a;
 cout<<pa -> f()<<endl;
 cout<<pa -> g()<<endl;
 pa=&c;
 cout<<pa -> f()<<endl;
 cout<<pa -> g()<<endl;
 return 0;
 }
```

OUTPUT:

1

2

1

5

b.

```
#include<iostream>
using namespace std;
class Base
{
int static i;
public:
Base(){
cout<<"base"<<endl;}
};
```

```
class Sub1: public virtual Base{};
class Sub2: public Base{};
class Multi: public Sub1, public
Sub2 {};
int main()
{
Multi m;
}
```

OUTPUT:

base

base

```
};
int main(){
A *pa;
A a;
```

```
};
int main(){
A *pa;
A a;
B b;
C c;
```

Question:2 **[5 points]**

Design an abstract base class **Flower** with the following characteristics: it has private member variables for name, color, and price. Define a constructor to initialize these variables. Declare pure virtual functions display() and storeToFile() to display flower information and store it in a file, respectively.

Implementing Derived Classes: Create three to two derived classes from Flower, each representing a different category of flowers (e.g. Lily, Sunflower).

For each derived class: Include additional attributes specific to that category (e.g., fragrance for Rose, petal count for Lily).

- Implement the display () function to display all attributes of that category.
- Implement the storeToFile() function to append flower information to a file named according to the flower category.

Testing and Storing Information: In the main() function, create instances of each derived class and demonstrate their functionality by displaying their information using the display() function. Ensure that the information about each flower is stored in a file named after its category.

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

// Abstract base class Flower
class Flower {
protected:
   string name;
   string color;
   double price;

public:
   Flower(string n, string c, double p) : name(n), color(c), price(p) {}
   virtual void display() const = 0;
   virtual void storeToFile() const = 0;
};


// Derived class Lily
class Lily : public Flower {
private:
   int petalCount;

public:
   Lily(string n, string c, double p, int pc) : Flower(n, c, p), petalCount(pc) {}
   void display() const override {
        cout << "Name: " << name << ", Color: " << color << ", Price: $" << price << ", Petal Count:
" << petalCount << endl;
   }
   void storeToFile() const override {
      ofstream file("Lily.txt", ios::app);
      if (file.is_open()) {
```

```
            file << "Name: " << name << ", Color: " << color << ", Price: $" << price << ", Petal Count:
" << petalCount << endl;
            file.close();
        }
    }
};


// Derived class Sunflower
class Sunflower : public Flower {
private:
    double height;

public:
    Sunflower(string n, string c, double p, double h) : Flower(n, c, p), height(h) {}
    void display() const override {
        cout << "Name: " << name << ", Color: " << color << ", Price: $" << price << ", Height: " <<
height << " inches" << endl;
    }
    void storeToFile() const override {
        ofstream file("Sunflower.txt", ios::app);
        if (file.is_open()) {
            file << "Name: " << name << ", Color: " << color << ", Price: $" << price << ", Height: "
<< height << " inches" << endl;
            file.close();
        }
    }
};

int main() {
    // Create instances of each flower category
    Lily lily("White Lily", "White", 4.25, 6);
    Sunflower sunflower("Yellow Sunflower", "Yellow", 2.99, 24.0);

    // Display flower information and store in files


    lily.display();
    lily.storeToFile();

    sunflower.display();
    sunflower.storeToFile();

    return 0;
}
```