ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Prof. P. Koumoutsakos, G. Tauriello
ETH Zentrum, CLT F 12, C 11
CH-8092 Zürich

Solution 5

Issued: week of 15.4.2013

In this exercise series, you will improve your background on the subjects of inheritance and polymorphism. In the following five questions, you will be asked to figure out the output of the programs as well as to provide modifications in order to achieve certain desired outputs.

Question 1:

Consider the following code.

```
#include <iostream>
class ODEsolver {
protected:
    void _rhs() {
        std::cout << "Euler rhs" << std::endl;
    }

    void _update() {
        std::cout << "Euler update" << std::endl;
    }

public:
    void step() {
        _rhs();
        _update();
    }
};

class ODEsolver_Leapfrog : public ODEsolver {
    void _update() {
        std::cout << "Leapfrog update" << std::endl;
    }
};
```

a) What would be the output of the following code?

```
ODEsolver * solver = new ODEsolver_Leapfrog;
solver->step();
```

The output is:
Euler rhs
Euler update

b) What would be the output of the following code?

```
ODEsolver_Leapfrog solver;
solver.step();
```

The output is:
```
Euler rhs
Euler update
```

c) Let us consider the same code again.

```
1 ODEsolver_Leapfrog solver;
2 solver.step();
```

In order to have the following output:

```
Euler rhs
Leapfrog update
Euler update
```

How would you change `ODEsolver` and `ODEsolver_Leapfrog` by **modifying only two lines**?

- line 8: add `virtual`
- line 21: replace with

```
1         std::cout<<"Leapfrog update"<<std::endl;
2         ODEsolver::_update();
```

Question 2:

Consider the following class.

```
1 class FinalClass  {
2 private:
3     FinalClass() { }
4 public:
5     static FinalClass* create() {
6         return new FinalClass();
7     }
8 };
```

a) Write the necessary code to obtain an instance of `FinalClass`.

```
FinalClass * p = FinalClass::create();
```

b) Consider the following class.

```
1 class DerivedClass: public FinalClass { };
2
3 int main() {
4     DerivedClass dc;
5     return 0;
6 }
```

Does the code compile? If not, what is the problem?

It does not compile. The derived class cannot be instantiated since it requires the constructor of FinalClass which is private.

c) What special feature does the class `FinalClass` exhibit?

`FinalClass` cannot be inherited (but it can be instantiated).

Question 3:

You work for the UN and you are compiling a crucial list of the species on the brink of extinction, defining whether hunting will be allowed or not. Consider the following code:

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

class Animal
{
public:
    string name;
    Animal(string name):name(name){}
    string okToBeHunted(){ return "allowed"; }
};

class Chicken : public Animal
{
public:
    Chicken():Animal("Chicken"){}
    string okToBeHunted(){ return "allowed"; }
};

class Sheep : public Animal
{
public:
    Sheep():Animal("Sheep"){}
    string okToBeHunted(){ return "allowed"; }
};

class SilkySifaka : public Animal
{
public:
    SilkySifaka():Animal("Silky Sifaka"){}
    string okToBeHunted(){ return "NOT allowed"; }
};

int main()
{
        Animal * animals[3];

        animals[0] = new Chicken;
        animals[1] = new Sheep;
        animals[2] = new SilkySifaka;

        for(unsigned int i=0; i<3; i++)
                cout << "The hunt of " << animals[i]->name
                        << " is " << animals[i]->okToBeHunted() <<
                           endl;

        for(unsigned int i=0; i<3; i++) delete animals[i];

        return 0;
}
```

a) What is the output of the code below on your terminal?
The print out of your program looks like:

```
The hunt of Chicken is allowed
The hunt of Sheep is allowed
The hunt of Silky Sifaka is allowed
```

b) Are you going to save the Silky Sifaka? And if not, what should you change in the code in order to print out `"NOT allowed"`?

You must declare the method `Animal::okToBeHunted()` as virtual:

```
#include <iostream>
#include <string>

using namespace std;

class Animal
{
public:
    string name;
    Animal(string name):name(name){}
    virtual string okToBeHunted(){ return "allowed"; }
};

class Chicken : public Animal
{
public:
    Chicken():Animal("Chicken"){}
    string okToBeHunted(){ return "allowed"; }
};

class Sheep : public Animal
{
public:
    Sheep():Animal("Sheep"){}
    string okToBeHunted(){ return "allowed"; }
};

class SilkySifaka : public Animal
{
public:
    SilkySifaka():Animal("Silky Sifaka"){}
    string okToBeHunted(){ return "NOT allowed"; }
};

int main()
{
        Animal * animals[3];

        animals[0] = new Chicken;
        animals[1] = new Sheep;
        animals[2] = new SilkySifaka;

        for(unsigned int i=0; i<3; i++)
                cout << "The hunt of " << animals[i]->name
                        << " is " << animals[i]->okToBeHunted() <<
                           endl;
```

```
46
47          for(unsigned int i=0; i<3; i++) delete animals[i];
48
49          return 0;
50  }
```

Question 4:

Consider the following code:

```
1  #include <iostream>
2
3  using namespace std;
4
5  class A {
6  public:
7      int field1, field2;
8      A(int v1, int v2  = 10) :field1(v1), field2(v2){}
9      virtual void print_fields() {
10         cout << "print A " << field1 << " " << field2 << endl;
11     }
12 };
13
14 class B: public A {
15 public:
16     B(int v1, int v2  = 10) :A(v1,v2){}
17     void print_fields() {
18         cout << "print B " << field1 << " " << field2 << endl;
19     }
20 };
21
22 class C: public A {
23 public:
24     C(int v1, int v2  = 10) :A(v1, v2){}
25     void print_fields() {
26         cout << "print C " << field1 << " " << field2 << endl;
27     }
28 };
```

a) What is the output of the following program? (choose one from below)

```
1  A a1(2);
2  A a2(2,11);
3  a1.print_fields();
4  a2.print_fields();
```

i) compile error on line 1

ii) compile error on line 2

iii) print A 2 10
print A 2 11

iv) print A 2 0
print A 2 11

The answer is iii)

```
print A 2 10
print A 2 11
```

b) What is the output of the following program? (choose one from below)

```
1  A a(1,1);
2  B b(1,1);
3  C c(1,1);
4  a.print_fields();
5  b.print_fields();
6  c.print_fields();
7
8  cout << endl;
9  A * a1 = &a;
10 A   a2 = b;
11 A * a3 = &c;
12 a1->print_fields();
13 a2.print_fields();
14 a3->print_fields();
```

i)
```
print A 1 1
print B 1 1
print C 1 1

print A 1 1
print A 1 1
print C 1 1
```

ii)
```
print A 1 1
print B 1 1
print C 1 1

print A 1 1
print B 1 1
print C 1 1
```

iii)
```
print A 1 1
print A 1 1
print A 1 1

print A 1 1
print B 1 1
print C 1 1
```

iv)
```
print A 1 1
print B 1 1
print C 1 1

print A 1 1
print A 1 1
print A 1 1
```

The answer is i)

```
print A 1 1
print B 1 1
print C 1 1

print A 1 1
print A 1 1
print C 1 1
```

c) Assume that you would remove the virtual keyword for the method `print_fields` in class `A`. Which of the choices given in question ?? would result as output when executing the code given there?

The answer is iv)

```
print A 1 1
print B 1 1
print C 1 1

print A 1 1
print A 1 1
print A 1 1
```

Question 5:

Given the following code:

```
#include <iostream>

class Animal
{
public:
    void printInfo()
    {
        std::cout << "Animal\n";
    }
};

class Bird : public Animal
{
public:
    virtual void printInfo()
    {
        std::cout << "Bird\n";
    }
};

class Eagle : public Bird
{
public:
    void printInfo()
    {
        std::cout << "Eagle\n";
    }
};

```

```
30 int main()
31 {
32     Animal animal;
33     Bird bird;
34     Eagle eagle;
35
36     Animal * a;
37     Bird * b;
38
39     a = &animal;
40     a->printInfo();
41
42     a = &bird;
43     a->printInfo();
44
45     a = &eagle;
46     a->printInfo();
47
48     b = &bird;
49     b->printInfo();
50
51     b = &eagle;
52     b->printInfo();
53
54     return 0;
55 }
```

---

a) What is the output?

Animal
Animal
Animal
Bird
Eagle

b) We now remove the `virtual` keyword from the method `printInfo` in the class `Bird` (line 15) and add it to the method in class `Animal` (line 6) instead. What is the output now?

Animal
Bird
Eagle
Bird
Eagle