NetID: hxie13 QuizID: 87105 Score: 2 / 4 Answer Source: PrairieLearn

1. Consider the following class definitions:

```
class Restaurant{
public:
    int rate() const;
private:
    double rating;
};

class Chipotle: public Restaurant {
public:
    int rateBad();
};
```

Where could the assignment `rating = 3.0;` appear for the private variable `rating`?

A. [Your Answer] `rate()` can make the assignment, but `rateBad()` cannot.
B. The answer to this question cannot be determined from the given code.
C. [Correct Answer] Neither `rate()` nor `rateBad()` can make the assignment.
D. Both `rate()` and `rateBad()` can make the assignment.
E. `rateBad()` can make the assignment, but `rate()` cannot.

2. Suppose class `sport` contains exactly one pure virtual function called `getDomain` and that class `volleyball` is a `public sport` that implements `getDomain`.

Which of the following C++ statements will certainly result in a compiler error? Make sure to read **all** options carefully.

A. `volleyball a; a.getDomain();`
B. [Correct Answer] [Your Answer] More than one of these will result in a compiler error.
C. `sport * a = new sport(); a->getDomain();`
D. `sport a; a.getDomain();`
E. It is possible that none of these will result in a compiler error.

3. What will be the output of the following program?

```
class One {
    public:
        virtual ~One() { cout << "One "; }
};

class Two : public One {
    public:
        Alpha *a2;
        Two() { a2 = new Alpha(); }
        virtual ~Two() { delete a2; cout<< "Two "; }
};

class Alpha {
   public:
       ~Alpha() { cout << "Alpha "; }
};

int main() {
    One* b = new Two;
    delete b;
}
```

A. `"Two Alpha One "`
B. [Your Answer] `"Alpha Two "`
C. [Correct Answer] `"Alpha Two One "`
D. `"One Alpha Two "`
E. `"One "`

4. What will be the output of the following program?

```
class Base {
    public:
        ~Base() {cout << "Destructing Base"; }
};

class Derived : public Base {
    public:
        ~Derived() { cout<< "Destructing Derived"; }
};

int main() {
    Base* b = new Derived;
    delete b;
}
```

A. Compiler error
B. [Correct Answer] [Your Answer] `"Destructing Base"`
C. None of the above
D. `"Destructing BaseDestructing Derived"`
E. `"Destructing Derived"`