

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

2. Consider the following class definitions:

```
class Sport{
public:
    virtual int winner();
private:
    int score;
};

class Volleyball: public Sport {
public:
    int loser();
};
```

Where could the assignment `score = 20;` appear for the private variable `score`?

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

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

```
class One {
public:
    Alpha *a1;
    One() { a1 = new Alpha(); }
    virtual ~One() { cout << "One "; delete a1; }
};

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

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

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

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

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