

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

Time Allowed: 30 minutes.

Total Points: 10

TYPE B

Question:1

a) Fix the errors [If any] in the given code below and predict the output

[2 points]

```
#include <iostream>
```

```
using namespace std;
```

```
class A {
```

```
public:
```

```
    A(int x) { cout << "A's Constructor: " << x << endl; }
```

```
};
```

```
class B : virtual public A {
```

```
public:
```

```
    B(int x) : A(x) { cout << "B's Constructor\n"; }
```

```
};
```

```
class C : virtual public A {
```

```
public:
```

```
    C(int x) : A(x) { cout << "C's Constructor\n"; }
```

```
};
```

```
class D : public B, public C {
```

```
public:
```

```
    D(int x) : A(x), B(x), C(x) {
        cout << "D's Constructor\n";
    }
```

```
};
```

```
int main() {
```

```
    D obj(10);
```

```
    return 0;
```

```
}
```

b) Encircle the correct statement

[2 points]

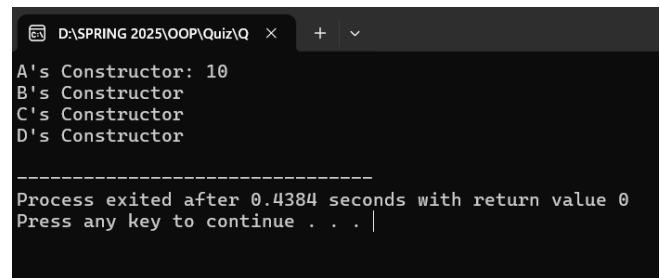
What happens when a class is inherited using **protected mode**?

a) Public members become private in the derived class

b) Public and protected members remain protected

c) All members become public in the derived class

d) The class cannot be inherited



```
D:\SPRING 2025\OOP\Quiz\Q x + v
A's Constructor: 10
B's Constructor
C's Constructor
D's Constructor

-----
Process exited after 0.4384 seconds with return value 0
Press any key to continue . . .
```

Question:2

[6 points]

Your task is to design a **Social Media Account Management System**.

In which:

- **Base Class: SocialMediaUser**

- **Protected Members:** string username → User's name, int postsMade → Number of posts made in the current session, int totalPostsMade → Total number of posts ever made

- **Methods:**
 - o virtual void createPost(string content) → A general post function.
 - o void displayTotalPosts() → Displays total number of posts.
 - o void createPost(string content, string mediaType) → Allows posting with a media type (e.g., image, video).
- **Derived Classes:**
- **RegularUser** (inherits from SocialMediaUser):
 - o **Redefines createPost(string content)** → Can only post text updates (max **200** characters).
- **InfluencerUser** (inherits from SocialMediaUser):
 - o **Redefines createPost(string content)** → Can post both **text and media** with priority visibility.

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

// Base Class
class SocialMediaUser {
protected:
    string username;
    int postsMade;
    int totalPostsMade;

public:
    SocialMediaUser(string name) {
        username = name;
        postsMade = 0;
        totalPostsMade = 0;
    }

    // Virtual method for general post
    virtual void createPost(string content) {
        cout << username << " posted: " << content << endl;
        postsMade++;
        totalPostsMade++;
    }

    // Overloaded method for media posts
    void createPost(string content, string mediaType) {
        cout << username << " posted with media (" << mediaType << "): " <<
content << endl;
        postsMade++;
        totalPostsMade++;
    }

    void displayTotalPosts() {
```

```

        cout << username << " has made a total of " << totalPostsMade << "
posts." << endl;
    }
};

// Derived Class - Regular User
class RegularUser : public SocialMediaUser {
public:
    RegularUser(string name) : SocialMediaUser(name) {}

    // Only text posts up to 200 characters
    void createPost(string content) override {
        if (content.length() <= 200) {
            cout << username << " (Regular User) posted: " << content <<
endl;
            postsMade++;
            totalPostsMade++;
        } else {
            cout << "Error: Regular users can only post text up to 200
characters." << endl;
        }
    }
};

// Derived Class - Influencer User
class InfluencerUser : public SocialMediaUser {
public:
    InfluencerUser(string name) : SocialMediaUser(name) {}

    // Can post with priority visibility
    void createPost(string content) override {
        cout << username << " (Influencer - Priority Post) posted: " <<
content << endl;
        postsMade++;
        totalPostsMade++;
    }
};

// Main function to test
int main() {
    RegularUser user1("Alice");
    InfluencerUser user2("Bob");

    user1.createPost("This is a simple text post from a regular user.");
    user1.createPost("This is a very long post that exceeds the character
limit for regular users. "

```

```
        "It should trigger an error message to show that
character limits are respected.");

    user2.createPost("Hey everyone! Check out my new vlog!");
    user2.createPost("Behind the scenes", "video");

    user1.displayTotalPosts();
    user2.displayTotalPosts();

    return 0;
}
```