1. What will be the output of the following code snippet?

```
#include <iostream>
class A {
public:
   A() { std::cout << "Constructor called"; }
};

int main() {
   A obj;
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "Constructor called" will be printed
d) No output

2. What is the output of the following code snippet?

```
#include <iostream>
class A {
public:
   A() { std::cout << "A "; }
};
class B : public A {
public:
   B() { std::cout << "B "; }
};
int main() {
   B obj;
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A B" will be printed
d) "B A" will be printed

3. What will be the output of the following code snippet?

```
#include <iostream>
class A {
public:
   virtual void foo() { std::cout << "A "; }
};
class B : public A {
public:
   void foo() override { std::cout << "B "; }
};
```

```
int main() {
    A* ptr = new B();
    ptr->foo();
    delete ptr;
    return 0;
}
```
a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

4. What is the output of the following code snippet?
```
#include <iostream>
class A {
public:
    A() { std::cout << "A "; }
    virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
    B() { std::cout << "B "; }
    ~B() { std::cout << "~B "; }
};

int main() {
    A* ptr = new B();
    delete ptr;
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A B ~B ~A" will be printed
d) "A B ~A" will be printed

5. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
    A() { std::cout << "A "; }
    virtual void foo() { std::cout << "A "; }
};
```

```
class B : public A {
public:
   B() { std::cout << "B "; }
   void foo() override { std::cout << "B "; }
};

int main() {
   A* ptr = new B();
   ptr->foo();
   delete ptr;
   return 0;
}
```

a) Compilation error
b) “A B B”
c) "A A" will be printed
d) "B B" will be printed

Answer: d) "B B" will be printed

6. What will be the output of the following code snippet?

```
#include <iostream>
class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   A obj;
   B& ref = dynamic_cast<B&>(obj);
   ref.foo();
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

7. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   A* ptr = new B();
   B* derived = dynamic_cast<B*>(ptr);
   derived->foo();
   delete ptr;
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

8. What will be the output of the following code snippet?

```
#include <iostream>
class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   A* ptr = new B();
   B& ref = static_cast<B&>(*ptr);
   ref.foo();
   delete ptr;
   return 0;
```

```
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

9. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   B obj;
   A* ptr = &obj;
   B& ref = dynamic_cast<B&>(*ptr);
   ref.foo();
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

10. What will be the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};
```

```
class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   B obj;
   A* ptr = &obj;
   B* derived = static_cast<B*>(ptr);
   derived->foo();
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

11. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   A* ptr = new B();
   B* derived = static_cast<B*>(ptr);
   derived->foo();
   delete ptr;
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

12. What is the output of the following code snippet?

```
#include <iostream>
class A {
public

:
   A() { std::cout << "A "; }
   virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
   B() { std::cout << "B "; }
   ~B() { std::cout << "~B "; }
};

int main() {
   B* ptr = new B();
   delete ptr;
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A B ~B ~A" will be printed
d) "B ~B ~A" will be printed
e) None of the above

13. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   A() { std::cout << "A "; }
   virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
   B() { std::cout << "B "; }
   ~B() { std::cout << "~B "; }
```

```
};

int main() {
    A* ptr = new B();
    delete ptr;
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A B ~B ~A" will be printed
d) "A ~A" will be printed

14. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
    virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
    void foo() { std::cout << "B "; }
};

void someFunction(A& obj) {
    obj.foo();
}

int main() {
    B obj;
    someFunction(obj);
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

15. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

void someFunction(A* ptr) {
   ptr->foo();
}

int main() {
   B obj;
   someFunction(&obj);
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

16. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   B obj;
   A& ref = obj;
   ref.foo();
```

```
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

17. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
    virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
    void foo() { std::cout << "B "; }
};

void someFunction(A& obj) {
    obj.foo();


}

int main() {
    B obj;
    A& ref = obj;
    someFunction(ref);
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

18. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

void someFunction(A* ptr) {
   ptr->foo();
}

int main() {
   B obj;
   A& ref = obj;
   someFunction(&ref);
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

19. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

void someFunction(A* ptr) {
   B& ref = dynamic_cast<B&>(*ptr);
   ref.foo();
```

```
}

int main() {
    A obj;
    someFunction(&obj);
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

20. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
    virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
    void foo() { std::cout << "B "; }
};

void someFunction(A& obj) {
    B& ref = dynamic_cast<B&>(obj);
    ref.foo();
}

int main() {
    A obj;
    someFunction(obj);
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

21. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
   virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
   ~B() { std::cout << "~B "; }
};

void someFunction(A& obj) {
   obj.foo();
}

int main() {
   B obj;
   someFunction(obj);
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

22. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
   virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
```

```
    ~B() { std::cout << "~B "; }
};

void someFunction(A* ptr) {
    ptr->foo();
}

int main() {
    B* ptr =

 new B();
    someFunction(ptr);
    delete ptr;
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

23. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
    virtual void foo() { std::cout << "A "; }
    virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
    void foo() { std::cout << "B "; }
    ~B() { std::cout << "~B "; }
};

void someFunction(A* ptr) {
    B* derived = dynamic_cast<B*>(ptr);
    derived->foo();
}

int main() {
    A* ptr = new B();
    someFunction(ptr);
```

```
    delete ptr;
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

24. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
    virtual void foo() { std::cout << "A "; }
    virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
    void foo() { std::cout << "B "; }
    ~B() { std::cout << "~B "; }
};

void someFunction(A* ptr) {
    B& ref = dynamic_cast<B&>(*ptr);
    ref.foo();
}

int main() {
    A obj;
    someFunction(&obj);
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

25. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
   virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
   ~B() { std::cout << "~B "; }
};

void someFunction(A* ptr) {
   B* derived = static_cast<B*>(ptr);
   derived->foo();
}

int main() {
   A* ptr = new A();
   someFunction(ptr);
   delete ptr;
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

26. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
   virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
```

```
    ~B() { std::cout << "~B "; }
};

void someFunction(A& obj) {
    B& ref = dynamic_cast<B&>(obj);
    ref.foo();
}

int main() {
    A* ptr = new A();
    someFunction(*ptr);
    delete ptr;
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

27. What is the output of

the following code snippet?

```
#include <iostream>

class A {
public:
    virtual void foo() { std::cout << "A "; }
    virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
    void foo() { std::cout << "B "; }
    ~B() { std::cout << "~B "; }
};

void someFunction(A& obj) {
    B* derived = static_cast<B*>(&obj);
    derived->foo();
}

int main() {
    A obj;
```

```
    someFunction(obj);
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

28. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
    virtual void foo() { std::cout << "A "; }
    virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
    void foo() { std::cout << "B "; }
    ~B() { std::cout << "~B "; }
};

void someFunction(A& obj) {
    B& ref = static_cast<B&>(obj);
    ref.foo();
}

int main() {
    B obj;
    someFunction(obj);
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

29. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   B obj;
   A& ref = obj;
   B& derived = dynamic_cast<B&>(ref);
   derived.foo();
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

30. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   B obj;
   A& ref = obj;
   B& derived = static_cast<B&>(ref);
   derived.foo();
```

```
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

31. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
    virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
    void foo() { std::cout << "B "; }
};

int main() {
    B obj;
    A* ptr = &obj;
    B* derived = dynamic_cast<B*>(ptr);
    derived->foo();
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

32. What is the output of the following code snippet?

```
#include <iostream>

class A {
```

```
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
};

int main() {
   B obj;
   A* ptr = &obj;
   B* derived = static_cast<B*>(ptr);
   derived->foo();
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

33. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
   virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
   void foo() { std::cout << "B "; }
   ~B() { std::cout << "~B "; }
};

int main() {
   B* ptr = new B();
   delete ptr;
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A B ~B ~A" will be printed
d) "B ~B ~A" will be printed

34. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   A() { std::cout << "A "; }
   virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
   B() { std::cout << "B "; }
   ~B() { std::cout << "~B "; }
};

int main() {
   A* ptr = new B();
   delete ptr;
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A B ~B ~A" will be printed
d) "A ~A" will be printed

35. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   A() { std::cout << "A "; }
   virtual ~A() { std::cout << "~A "; }
};

class B : public A {
public:
```

```
    B() { std::cout << "B "; }
    ~B() { std::cout << "~B "; }
};

int main() {
    B* ptr = new B();
    delete ptr;
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A B ~B ~A" will be printed
d) "B ~B ~A" will be printed

36. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
    virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
    void foo() override { std::cout << "B "; }
};

int main() {
    B obj;
    A* ptr = &obj;
    ptr->foo();
    return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

37. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo

() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() override { std::cout << "B "; }
};

int main() {
   B obj;
   A* ptr = &obj;
   static_cast<B*>(ptr)->foo();
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

38. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() override { std::cout << "B "; }
};

void someFunction(A& obj) {
   obj.foo();
}
```

```
int main() {
   B obj;
   someFunction(obj);
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

39. What is the output of the following code snippet?

```
#include <iostream>

class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() override { std::cout << "B "; }
};

void someFunction(A* ptr) {
   ptr->foo();
}

int main() {
   B obj;
   someFunction(&obj);
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed

40. What is the output of the following code snippet?

```
#include <iostream>
```

```
class A {
public:
   virtual void foo() { std::cout << "A "; }
};

class B : public A {
public:
   void foo() override { std::cout << "B "; }
};

void someFunction(A* ptr) {
   B& ref = dynamic_cast<B&>(*ptr);
   ref.foo();
}

int main() {
   A obj;
   someFunction(&obj);
   return 0;
}
```

a) Compilation error
b) Runtime error
c) "A" will be printed
d) "B" will be printed
Answer: b) Runtime error The only answer I provide