Question 1 of 10 1.0 Points

Assume that there are 2 members, field1 and foo, declared in class A and there are also 2 members, field2 and fie, declared in class B, and A is the superclass of B, how many members in class A and class B?

- A. A has 4 members while B has 2 members
- B. A and B has 2 members
- C. A and B has 4 members
- D. A has 2 members while B has 4 members

Reset Selection

Question 2 of 10

Assume that A is the super class of B and C, and variables x, y, z are declared as follows,
A x;
B y;
C z;
Select valid assignment statement(s)

- [x] A. x = new B;
- [ ] B. y = new A;
- [ ] C. z = new B;
- [ ] D. z = new A;
- [x] E. x = new C;

Question 3 of 10 1.0 Points

Assume that class A is the super class of class B and class B is the super class of class C. There is method foo declared in class A and it is overriden in class B and also class C (i.e. foo is declared in B and C). Variables x, y, z are declared as follows,
A x;
B y;
C z;
Match the call to the set of its targets.

A. {foo in C}
B. {foo in A, foo in B, foo in C}
C. {foo in A, foo in B}
D. {foo in B, foo in C}

B 1. x.foo()
D 2. y.foo()
A 3. z.foo()
C 4. Not match

Question 4 of 10

Assume that class A is the super class of class B, and class B is the super class of class C. There are declarations of static method foo in class A, B and C.

- A. B.foo() may call just foo defined in B
- B. A.foo() may call foo defined in A, in B or in C
- C. B.foo() may call foo defined in B or in C
- D. A.foo() may call just foo defined in A
- E. B.foo() may call foo defined in A or in B

Question 5 of 10

To declare a static field *sfield* for class ABC in Scala, how must programmers write?

A. Add the following line in object ABC,

static var sfield:Int

B. Add the following line in class ABC,

var sfield:Int

C. Add the following line in object ABC,

var sfield:Int

D. Add the following line in class ABC,

static var sfield:Int

Question 6 of 10

In Scala, to declare class ABC as a subclass of class DEF, how must programmers write?

A. class ABC: DEF

B. class DEF extends ABC

C. class ABC extends DEF

D. class DEF super ABC

Reset Selection

Question 7 of 10

Given the following code in Scala,

```
abstract class A {     def cal(x:Int):Int  }
trait B extends A  { abstract override def cal(x:Int):Int = super.cal(x * 2)}
trait C extends A {abstract override def cal(x:Int):Int = super.cal(x + 1)}
class D extends A {def cal(x:Int):Int = x}
val t = new D with B with C
println(t.cal(5))
```

What is the printed value?

A. 5

B. 11

C. 12

D. Other value

Reset Selection

v

Question 8 of 10

How to define a member which can only be accessed inside the object? For example,
class X {
    var f:Int // <= how to declare this field to be accessed just inside an object of X
    var m = new X
    //this.f is Ok but m.f is wrong in the code of class X
}

- ○ A. var f:Int
- ○ B. protected[this] var f:Int
- ○ C. protected var f:Int
- ○ D. private var f:Int
- ◉ E. private[this] var f:Int

Reset Selection

Question 9 of 10

How to create an object of a case class in Scala? For example, for the following case class,
case class Rational(n:Int,d:Int)
Please select the shortest correct answer.

- ◉ A. Rational(2,3)
- ○ B. object Rational(2,3)
- ○ C. new Rational(2,3)
- ○ D. create Rational(2,3)

Question 10 of 10

which is/are the declaration(s) of A such that the new A is INVALID?

- ☑ A. trait A
- ☐ B. case class A
- ☑ C. abstract class A
- ☐ D. class A